

AD-A133 892

COINCIDENT EXTRACTION OF LINE OBJECTS FROM STEREO IMAGE

1/2

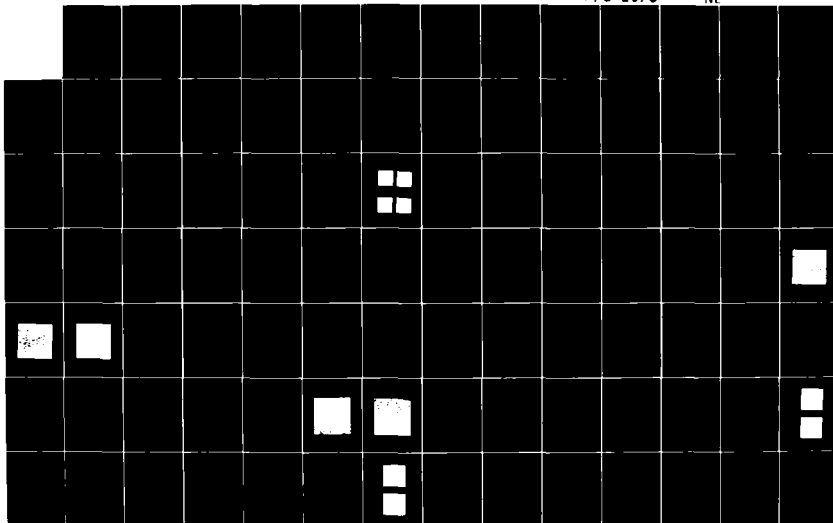
PAIRS(U) FORSCHUNGSINSTITUT FUER
INFORMATIONSVERRARBEITUNG KARLSRUHE (G..

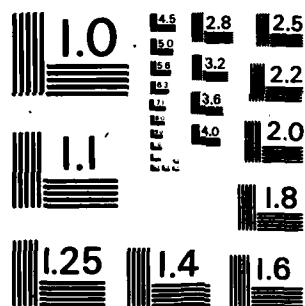
UNCLASSIFIED

H KAZMIERCZAK ET AL. SEP 83 FIM-115

F/G 20/6

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD A133 892

(1)

Coincident Extraction of Line Objects from Stereo Image Pairs

Final Technical Report

September 1983

EUROPEAN RESEARCH OFFICE
United States Army
London England

CONTRACT NUMBER DAJA 37-82-C-0243

FIM/FGAN, Breslauer Str. 48
D-7500 Karlsruhe 1, W. Germany

DTIC
ELECTE
OCT 24 1983
A

Approved for Public Release; distribution unlimited

Original contains color
plates: All DTIC reproductions
will be in black and
white

83 10 17 174

DTIC FILE COPY

Coincident Extraction of Line Objects from Stereo Image Pairs

Final Technical Report
by

H. Kazmierczak, M. Sties
R. Lubkowitz, M. Bohner

September 1983

EUROPEAN RESEARCH OFFICE
United States Army
London England

CONTRACT NUMBER DAJA 37-82-C-0243

**FIM/FGAN, Breslauer Str. 48
D-7500 Karlsruhe 1, W. Germany**

Principal Investigator: Prof. Dr.-Ing. H. Kazmierczak

FIM Report No. 115

The Research reported in this document has been made possible through the support and sponsorship of the U. S. Government through its European Research Office of the U. S. Army. This report is intended only for the internal management use of the Contractor and the U. S. Government.



Accession For

NIS 00001
F. TAN
H. A. T. T.
J. T. T. T.

DISP

A

zak

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

R&D 4095-R-EN

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. ADA133892	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Coincident Extraction of Line Objects from Stereo Image Pairs		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report Feb 82 - Sept 83
7. AUTHOR(s) H. Kazmierczak, M. Sties, R. Lubkowitz, M. Bohner		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Forschungsinstitut für Informationsverarbeitung und Mustererkennung, Breslauer Strasse 48, D-7500 Karlsruhe 1, West Germany		8. CONTRACT OR GRANT NUMBER(s) DAJA37-82-C-0243
11. CONTROLLING OFFICE NAME AND ADDRESS USARDSG-UK, Box 65, FPO New York, NY 09510		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102A-T161102-BH57-01
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE September 1983
		13. NUMBER OF PAGES 110
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Automatic extraction, software, aerial imagery, air photos, digital processing, computer programs, stereo image pairs <i>(e.g., objects are roads, rivers, railroads and multilane highways)</i>		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) A procedure for the automatic extraction of line objects from stereo pairs of aerial images has been developed and implemented on a DEC VAX 11/780 computer. The procedure and its main functions are described in summary and in detail. A detailed documentation of the various test results is included as well as assessment and conclusions.		

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Table of Contents

1. Statement of the problem
2. Separate line extraction from stereo images
3. The procedure of coincident line extraction
4. Details of the main functions
 - 4.1 Extraction procedure
 - 4.2 Maximum length polygon compilation
 - 4.3 Image registration algorithms
 - 4.4 Verification procedure
 - 4.4.1 Kernel of the verification procedure
 - 4.4.2 Backmapping and reinsertion of accepted polygons
 - 4.4.3 Reconstruction of intersections
 - 4.5 Final result processing
5. Presentation of the results
 - 5.1 FIM image processing system
 - 5.2 Extraction results in images of 8000 x 8000 pixels
 - 5.3 Results of coincident extraction in stereo image pairs of 4000 x 4000 pixels
 - 5.4 Explanation of details in segments of the stereo image pairs of 1000 x 1000 pixels
6. Discussion of the results and conclusions

1. Statement of the problem

Methods for an automatic extraction of line shaped objects (subsequently called line objects), from aerial images, e.g. objects such as roads, rivers, railroads and multilane highways, have been developed and documented /1,2/. This research project has been continued to apply the extraction methods coincidentally to stereo image pairs, the results of which are reported here.

USAETL provided transparencies of three aerial images labeled No. 52, 54 and 56, of 9" by 9" at a scale of approx. 1:70000. The images No. 52 and 54 as well as the images No. 54 and 56 form a stereo image pair each with approx. 60% overlapping area in each pair. A small strip of the area imaged is contained in all three images.

The problem we faced can be summarized as follows:

Produce extraction results of identical line objects from both images of a stereo pair.

This would yield the coordinates of corresponding locations in the overlapping part of a stereo pair by application of image analysis techniques, but without a systematic and total processing of the image matrices and without subimage correlation. In our solution of the problem, the extraction methods do not process both images simultaneously; the extraction is performed subsequently in both images, always taking the extraction result from one image as a local guidance for the extraction process in the other image. To achieve this goal, the existing software system had to be modified and extended considerably.

The following sections of this report will give a detailed description of the procedure of coincident line extraction and will address the main functions of this procedure in detail. Sections 5 and 6 of this report will be devoted to document and explain the results achieved.

2. Separate line extraction from stereo images

Reference is made to the Third Progress Report /3/. The transparencies of all three aerial images provided by USAETL were digitized by an OPTRONICS Scanner C4500 to produce image matrices of 9201 x 9201 pixels each. The parameters of the scan were

raster width	25 μ m (0.001 inch)
dot size	25 μ m (0.001 inch, corresponding to a circular spot size of approx. 2 m diameter on earth)
density	2 D
A/D conversion	log, 8 bit.

In order to get rid of some inconvenient imaging distortions at the image boundaries we selected the central area of 7968 x 7968 pixels of each image for our test purposes. The number 7968 comes from 8 x 1024 pixels, subtracting 7 x 32 pixels overlapping at the subimage boundaries.

The central image No. 54 of the sequence of overlapping images, which had been used as a test image during the previous research project /1,2/, had to be digitized again to guarantee the same dynamic range of the intensity values of all three images to be used in this experiment. The digitization of this image which took place two years ago had been performed on another scanning device.

Any processing operation of the image data had to be confined to image segments of 1024 x 1024 pixels because of the limited size of the working memory. Thus we segmented all three image matrices into three mosaics of 8 x 8 subimages each with an overlapping of 32 pixels between any two adjacent subimages of a mosaic.

This change of the subimage size from 512 x 512 to 1024 x 1024 pixels in this project caused some minor modifications of the line extraction software. To verify the correct performance of the extraction software, we applied the extraction to each of the three images separately and independently. Due to the bigger subimage size the extraction process performed faster than before. The results of this experiment will be presented and explained in section 5 of this report.

3. The procedure of coincident line extraction

During the development of the procedure of coincident line extraction we realized that we could no longer operate in a fixed mosaic of 8 x 8 subimages of each image. As we had to transfer extraction results from one image to the partner image of the stereo pair, we faced the problem of defining the subimage boundaries at corresponding locations in both images. If this would not be the case, the extraction results of a submatrix of one image would have covered locations of (in the worst case) four submatrices of the partner image which would have required an unnecessary amount of image data transfer from the disk storage to the working memory.

We decided to store all three image matrices unsegmented on disk and to work in a dynamic mosaic, i.e. processing again was confined to image segments of 1024 x 1024 pixels; however, the coordinates of an image segment were determined immediately after a loading request to allow loading of the appropriately located image segment.

An alternative experiment with virtual addressing within the limits of the 8000 x 8000 pixels of each image without explicit loading of subimages could not yet be performed due to computer system software problems.

The procedure of coincident extraction of line objects from a stereo image pair (subsequently called "left image" IL and "right image" IR) starts with an initial extraction from one image. There is no requirement to prefer one of both stereo images for the initial extraction process; however, we will explain the procedure by starting in IL. All extraction parameters are defaulted with respect to image scale and pixel size:

- type of contrast
- amount of contrast required
- upper limit of local gray level variation along a line object
- minimum width of a line object
- maximum width of a line object.

The automatic extraction process starts with a search for starting points and an iterative application of the local and the regional extraction method to the top left subimage of IL and works systematically through all subimages of 1024 x 1024 pixels of IL as it has been described for the previous project /1,2/. The local extraction method (LEM) has received an improved adaptivity by adding a parameter analysis capability. At each starting point the properties of that segment of the line object are analyzed and the values measured replace the respective default values (contrast, line width). LEM performs "conservatively", i.e. no continuation of a line object is accepted in cases of considerable distortions or property variations. When starting at a starting point location, LEM requires a few contiguous extraction steps along the line object in at least one of both directions to be acceptable, else the respective starting point will be deleted. In case of successful extraction, LEM proceeds incrementally until one of the following conditions causes it to stop:

- image boundary is reached; a final dead end is generated at that location
- subimage boundary is reached; a temporary dead end is gener-

ated at that location for later continuation in the adjacent subimage

- a location already processed is reached (known road location); a final dead end is generated at that location
- a distortion of the line is reached which is severe enough to allow no further continuation of LEM (see /1/ for details); a temporary dead end is generated at that location for later continuation by the regional extraction method.

The results so far will be:

- a string of coordinates of vertices of polygons approximating the extracted lines or line segments. The polygon vertices are stored in a sequential file on a "first come first store" basis, i.e. the polygon segments are not sorted locally. This file is called PLIILCL which indicates that the polygons represent lines extracted from the left image of the stereo pair and the coordinates of these polygons are related to the coordinate system of the left image
- a result matrix of a size identical to the image matrix. The result matrix displays a binary raster representation of all lines or line segments extracted. To be more precise, one byte is provided for each element of the result matrix to describe the extraction result by ASCII symbols (symbol S for starting points and symbol * for the extraction results of LEM)
- several lists of dead end locations; the final dead ends are separated from the set of temporary dead ends at subimage boundaries and also separated from the temporary "break off" dead ends of LEM.

When LEM has stopped the extraction process in the subimage, the regional extraction method (REM) will continue the extraction at all break off dead ends of LEM, if there are any. REM complements LEM, because REM can accept a higher degree of distortions (see /1/ for details). REM continues the extraction until one of the following conditions causes it to stop:

- image boundary is reached (final dead end)
- subimage boundary is reached (temporary dead end to be used in the adjacent subimage)
- location already processed is reached (final dead end at known road location)
- a distortion of the line is reached, which is severe enough to allow no further extraction by REM (final break off dead end)
- a location is reached where the line object has "ideal" properties complying totally with the standard line description parameter values in LEM; a temporary dead end is generated at that location for later continuation by LEM which performs faster.

When REM has stopped after exhausting its input list of temporary dead ends, the following results will have been added to the results already existing:

- polygon vertices are added to the vector representation of the results PLILCL, again on the basis of "first come first store"
- ASCII symbols + are inserted at each element of the result matrix which is covered by the polygon segments representing the lines extracted by REM
- final dead ends, temporary dead ends at subimage boundaries, final break off dead ends and temporary dead ends at ideal line segments are added to the respective files already existing or newly generated.

The iterative application of starting point search, LEM and REM continues until all extractable lines have been found in the actual subimage. The same procedure is applied to the next subimage; however, before searching for starting points in that subimage, the relevant entries in the file of subimage boundary dead ends is exhausted for possible continuations of lines extending from the previously processed subimage into the actual one. The same sequence of extraction steps is performed when starting from these dead ends, as it has been explained for the case of starting from starting points. This process continues through the sequence of 64 subimages of 1024 x 1024 pixels each. A second run through the sequence of subimages is initialized. This time, only those subimages will be loaded and processed, which have valid entries in the file of subimage boundary dead ends. These entries may result from lines pointing back to previously processed subimages already processed during the first extraction sequence. A third and further runs through the sequence of subimages may follow, as long as subimage boundary dead ends are generated which refer back to previously processed subimages. The search for starting points is not applied during the second and every further extraction run through the subimage sequence. In our practical tests we never noticed more than three runs through the subimage sequence; furthermore a rapidly de-

creasing number of subimages is affected by the second and third run (in one example, only two subimages out of the total of 64 were affected during the third run).

At the end of this initial line extraction from the left image the following results will have been accumulated:

- a file PLIILCL containing the coordinates of the polygon vertices. It has already been mentioned that this vector representation of the extraction result is not sorted locally, but has been accumulated "piece by piece" according to the data dependent proceeding of the extraction methods through the image matrix
- a result matrix related to the left image (called RMIL) the size of which is identical to the size of the image matrix itself. RMIL contains the raster representation of the extraction result, encoded by the symbols S for starting points, * respectively + for the matrix elements which are covered by the polygon elements
- a file containing dead end coordinates (image boundary dead ends, final break off dead ends). The dead end file is also not sorted locally.

We are aware of the fact, that one of both the raster and the vector representation of the extraction results is redundant, because there exist standard conversion procedures between both representations. However, for reasons of computational speed and ease of data management and control, we retain both representations simultaneously.

At the end of this initial line extraction (from the left image of the stereo pair, as we assumed for the purpose of explanation), neither the extraction results nor the dead ends can be called "final", because we have to check all these resulting data by evaluating the stereo partner image. Only after accepting extraction results for identical line objects from both images we will call them final results and the related break off dead ends will be called final dead ends. Thus, with reference to the initial extraction described above, the term "final" has to be replaced by the term "preliminary final".

The next stage of the process of coincident extraction addresses the verification and possible continuation of the initial extraction results from the left image by proper processing of the right image. Before turning to the verification process we have to describe some preparatory processing steps. The verification process requires that the extraction results of the left image be mapped into the coordinate system of the right image, because we use them as a local guidance for the evaluation of the right image. As the coordinate transformation of the complete result matrix RMIL is an extremely time consuming action, we decided to submit the vector representation of the results to this mapping function. However, in order to minimize requirements of data access to the right image, we had to apply a sorting function to the result file PLILCL to produce sequences of contiguous polygon segments of the maximum length possible. We considered two possible solutions of this problem:

- concatenating the various polygon sections contained in PLILCL by searching identical polygon vertices
- conversion of the raster representation contained in RMIL into vector format of maximum length chains.

Due to computational speed we chose the second solution which was implemented as a straightforward extraction process of the lines encoded by the symbols S , * and + from the result matrix. This ternary extraction process starts at every dead end location and proceeds along the chain of symbols in the result matrix until it stops at another dead end or at an intersection or branching of the symbol chains. Thus, this process of maximum length polygon compilation generates a file MLPILCL (maximum length polygons extracted from the left image and related to the coordinate system of the left image) the polygons of which extend either between two dead ends or two intersections (or branchings, subsequently called intersections only), or an intersection and a dead end or vice versa. Simultaneously, all symbols are deleted from the result matrix as they are processed during this operation. The reason for deletion is two-

fold: no extraction result can be accepted without verification; the accepted results will be reinserted into this result matrix later. Secondly we simply prevent multiple processing of a symbol by deleting it immediately. Thus, at the end of this processing step, the result matrix RMIL will be reset to zero completely.

The next processing step performs the mapping function of the extraction result contained in MLPILCL to the coordinate system of the stereo partner image (the right image IR in our example). Initially we wanted to apply the projective mapping function based on 4 registration control points. However, the identification of 4 control points in one plane of the 3D surface of the terrain was not possible with our instrumentation, as we were viewing the 2D images only. Each of a few different definitions of the set of 4 control points yielded a different misregistration which was visually checked by overlaying the right image and the mapped graphic line network MLPILCR (the maximum length polygons extracted from IL but related to the coordinate system of IR after registration). In some cases the misregistration was noticed to amount up to 60 pixels. As this result did not comply with the idea of local guidance of the evaluation process in IR, we decided to use another mapping function which yielded smaller amounts of misregistration. We selected a polynomial interpolation function (for details see section 4.3 of this report). Using this function we are able to influence the accuracy of the registration results through the number and the proper distribution of the control points. With a number of approx. 30 control points the computational requirements are only slightly higher than for the projective mapping function; however, the misregistration was reliably smaller than 12 pixels which suited our purpose completely.

After having registered the initial extraction result of IL to the coordinate system of IR to this degree of accuracy we turn to the very process of verification. This process is to serve two purposes:

- the successful verification in IR proves that the respective line object exists in both images of the stereo pair and that the respective extraction result can finally be accepted
- the verification will yield the true coordinates of the respective line object with regard to IR. Thus we will have true and reliable extraction results for corresponding line objects from both images of the stereo pair.

The verification process is performed by a modified version of REM: take a polygon segment of MLPILCR (the coordinates of which are related to IR) to predict the existence of a line object segment in IR, define a rectangular "area of interest" at that location in IR, perform the same operations as they have been described for the regional extraction method. The following modifications were implemented for REM to be applied for this purpose:

- the length of an AoI is dictated by the length of the polygon segments of MLPILCR. In case of extreme short or extreme long polygon segments, special handling had to be developed which is described in section 4.4.1 of this report
- the orientation and location of an AoI is defined to be symmetrical with respect to the polygon segment of MLPILCR; however, as we do not know the accuracy of the mapping function which produces the location of this polygon segment to predict the existence of a line in IR, we will not require this line (if it is detected and accepted after evaluation of the AoI at this location) to start at the center of the base line of the AoI. Here, we have to accept the maximum confidence evaluation result (provided that the confidence exceeds a lower limit) as extraction result for this part of IR. (When using REM in a normal extraction mode we prefer that evaluation result within an AoI which starts at the center of the AoI base line in order to ensure connection between the results of consecutive AoIs).

The result of this verification process at a location in IR predicted by a mapped polygon segment can be threefold:

- in most cases we succeed in verifying the respective line segment within that AoI. Now we are sure that this part of the line object exists in both IL and IR and we accept it finally and definitely. In consequence we generate a file of finally accepted polygon segments for IL called APLILCR, the vertex coordinates of which are taken from MLPILCR, and

a file of finally accepted polygon segments for IR called APLIRCR, the vertex coordinates of which are taken from this very verification result in IR, and we insert the ASCII symbol E at all elements of RMIR which are covered by that polygon segment in IR. The symbol "E" in the result matrix of IR stands for "external" which indicates that this line segment has initially been discovered in IL. The corresponding entry of symbol I (standing for "internal") into RMIL will also occur, but at a later processing step

- in some cases we do not succeed in verifying the suspected line segment at the predicted location in IR. This may be due to false extraction results from IL or noisy or distorted corresponding locations in IR which renders these parts of the line objects undetectable. Consequently, we do not accept the preliminary extraction results which came from IL, neither for IL nor for IR and no entries, neither to the respective files of accepted polygon segments nor to the result matrices RMIR and RMIL, will occur
- in some cases we succeed in verifying one or several parts of the suspected line segment at the predicted location in IR. We will accept the corresponding parts of the respective polygon segment by entering the proper coordinates into the files APLILCR and APLIRCR and the proper symbols into the result matrix RMIR. The result matrix RMIL will be updated in a later processing step. For a detailed explanation of this operation see section 4.4 of this report.

Regarding dead end locations the following two files will be generated:

- the file FDEIRCR containing final dead ends of the verification process. These are all those locations where the verification process failed to detect the predicted line objects in IR or where the lines exceeded the image boundaries. At these locations a continuing line object extraction will not be possible and will not be attempted, hence the name final dead end. For details see section 4.4 of this report
- the file VDEIRCR containing verification dead ends. These are those remaining open ends of the polygons accepted where a continuing line object extraction from IR seems to be possible.

The generation of the final dead end set for IL (file FDEILCL) is incorporated into the final processing step (see section 4.5 of this report).

This process of verification is performed for each segment of

MLPILCR. At the end of this process, the coordinates of the accepted polygon segments of APLILCR are mapped back to the coordinate system of IL using the inverse of the mapping function already explained. This yields a first part of the final result file APLILCL related to the left image IL.

A last process of this stage of the coincident extraction procedure attempts to restore line intersections which may have existed in MLPILCR, but could have been disrupted or separated by the verification process. This restoration operation is performed on the contents of RMIR by checking each open end of a polygon accepted (i.e. VDE dead end locations where the verification stopped due to missing prediction of continuation) whether any other polygon segment accepted is "nearby". The restoration of broken intersections will also be performed on the contents of RMIL starting from the respective dead ends related to IL; however, this will be performed at a later processing step, when we again are processing IL. For details of this process see section 4.4.3 of this report.

The next stage of the overall procedure pertains to the continuation of the line object extraction from IR. This is accomplished by starting the regular extraction process, i.e. the iterative application of LEM and REM, at each verification dead end of the polygons accepted in IR. At each dead end LEM adapts its parameters to the actual line properties and the iteration extraction process is performed. Each extraction result leads to entries into the respective result file PLIRCR and into the respective result matrix RMIR (symbols * and +). As this will be the first application of the extraction process to IR, the search for new starting points in IR will also be initialized and the subsequent extraction by LEM and REM will try to extract new line objects from IR, which had not yet been found in IL. If, at this point, any continuing or new line object or part of a line object has been extracted, the following results will have been produced:

- a file PLIRCR of complementary line segments, extracted from IR and related to the coordinate system of IR
- new entries of the symbols S , * and + into the result matrix RMIR
- a file containing dead end coordinates (image boundary, break off, known road), where the continuation of the extraction was not possible. (All of these dead ends will be called preliminary final, because they will be submitted to verification in IL. However, at locations, where no continuation of the extraction was found, a final dead end will be generated).

These results have to be verified in IL. Consequently, we apply the sorting function already described to the contents of PLIRCR to generate maximum length polygons (file MLPIRCR) of this extraction result. These polygons will then be mapped to the coordinate system of IL to produce the file MLPIRCL. Now, we are back again in IL and the loop is closed. Our first priority at this point will be the verification of the extraction results from IR. Before this operation takes place, two other tasks have to be solved:

- enter the symbol I to RMIL at each matrix element which is covered by the actual contents of APLILCL (extraction results from IL which have been verified successfully in IR)
- restore line intersections in this result which may have been disrupted or separated by the verification process. This operation is performed on the symbols I of RMIL by checking the vicinity of every dead end associated to the final result known so far.

The verification process which follows needs no further explanation: each polygon element of MLPIRCL initiates an AOI in IL to be evaluated with regard to accepting or rejecting this extraction result from IR. Result files (APLILCL and APLIRCL) and entries to RMIL (symbol E for external) are updated as described above. The last actions of this processing step will map the polygon segments of APLIRCL back to the coordinate system of IR to update APLIRCR and restore possibly disrupted intersections of the results.

Another iteration of the whole sequence of processing steps will follow: complementary extraction of line objects from IL (using only the verification dead ends to start and skipping the search for new starting points in IL), compilation of maximum length polygons of these extraction results, mapping of these polygons into the coordinate system of IR, verification of these extraction results, etc., until, after some steps, no significant complementary extraction result will be found. In most of our tests the fourth application of the extraction methods did not contribute significantly to the final results, so that the whole procedure could be stopped. There is, of course, no difficulty in stopping the processing at an earlier step or allowing more iterations than four. Any final stop should occur after a verification phase. Note, that back-mapping and reinsertion of results, related to the stereo partner image, will always be necessary.

At that point, the following results will exist:

- result matrices RMIL and RMIR containing symbols I and E . The symbols I represent results which have been extracted from the respective image and accepted by verification in the stereo partner image; the symbols E represent results which have been extracted from the stereo partner image and accepted by verification in the respective image
- result files APLILCL and APLIRCR containing the vector representation of the final extraction result (note, that no local sorting has been performed for the results of the different iterations of extraction and verification)
- files FDEILCL and FDEIRCR containing the locations of the final dead ends, the open ends of the resulting line network where the extraction ended either due to image boundary or due to noisy or distorted image data.

To produce a locally sorted, final vector representation of the result, a final maximum length polygon compilation is once more applied to both result matrices which yields the two result files FMLPILCL and FMLPIRCR (for details see section 4.5 of this report). These two files contain the final result in the most dense representation, and this will be the preferred

representation to be shipped to the contractor. The results of these two files are related to the respective image coordinate system. After properly mapping these two resulting line networks into one coordinate system, the remaining local deviations of the coordinates of corresponding locations will represent the parallaxes between the two stereo images.

4. Details of the main functions

4.1 Extraction procedure

Basically, the procedure of line object extraction from aerial images has remained unchanged with respect to the description contained in /1,2/. There are those three main modules already mentioned:

- method for the recognition of starting points
- local extraction method LEM
- regional extraction method REM

which are characterized by the following main features:

To detect starting points a regular pattern of lines and columns of the image matrix is selected as sample lines to be evaluated. In a first step, candidates for starting points are defined at those locations of the sample lines where the gray level profile satisfies the required properties. These candidate locations are submitted to evaluations more sophisticated to test whether contrast, uniformity, straightness, width, etc. comply with the required properties. Thus, in most cases we succeed in detecting some true segments of the line objects as reliable starting points for the extraction methods.

The local extraction method starts its evaluation at these starting points and follows the line object incrementally. By analyzing its current location, LEM predicts the orientation and the properties of a next segment of the line object (prediction of properties is restricted to the limits defined by the input parameters). This prediction will then be verified by analysis of a gray level profile. In case of compliance of the predicted properties with the properties measured this next increment of the line object is accepted. In case of non-compliance more sophisticated algorithms are applied to decide, whether the line object ends at that point, whether it continues with changed properties, whether the line object branches or intersects another or the image is distorted locally.

The regional extraction method complements the local extraction method: REM never starts the extraction at a starting point location; REM is applied to continue the task at locations where LEM stops due to distortion, noise, object occlusion, etc. REM works in a predetermined limited part of the image (prediction of the area to be processed). This area called "area of interest" usually has a rectangular shape. The area of interest is expected to contain completely an interesting part of the line object to be extracted.

In addition the orientation of the rectangle is supposed to comply with the predominant orientation of the object. It has been shown that a complete analysis of the 2D gray level matrix of the area of interest is not necessary to detect the continuation of a line object. The analysis of several 1D sample lines, which possibly contain cross-sections of the object, has proved to be sufficient.

The sample lines are perpendicular to the main axis of the rectangle. The first part of the line detection algorithm consists of a gray level analysis of the sample lines to dis-

cover candidates for object points. A confidence value is computed for each point of each sample line to indicate its probability of belonging to a line object. A semi-threshold operation is applied to this matrix of confidence values to retain only a few points with the highest values of each sample line.

The second part of the algorithm works on this set of candidates to detect subsets, the points of which are in a collinear position. This computation is done iteratively to increase the confidence value of a point, if collinear continuation points exist, or to decrease its confidence value otherwise. Thus it is possible to confirm all candidates, which contribute to form a line, while the more or less isolated candidates can be discarded. At the same time, two vectors have been computed for each confirmed candidate, which point to both neighbouring candidates in a line above and below. On the basis of this direction information and the last known object point P, the line following task is accomplished as the last part of this algorithm.

4.2 Maximum length polygon compilation

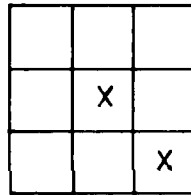
The maximum length polygon compilation is a special extraction method for quasi-binary images. Given the ends of a line network, it separates the network in polygons of maximal length, where a polygon of maximal length is defined as the longest string of succeeding polygon segments without an intersection. That means that a maximum length polygon ends either in a dead end or in an intersection (see figure 4.1). Figure 4.1a consists of four, figure 4.1b only of one maximum length polygon. Line AB of figure 4.1a ends at each side in an intersection, the polygon of figure 4.1b at both sides in a dead end.



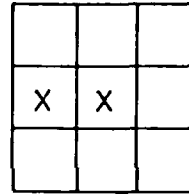
Fig. 4.1: Examples of maximum length polygons

In the process of coincident extraction of line objects from stereo image pairs the maximum length polygon compilation is used to sort the extracted lines. The extraction algorithm stores the lines found on a first come first store basis in the files PLIICL respectively PLIRCR. To verify the extracted lines it is much more efficient to use the reconstruction of connected polygons instead of the separate parts. To solve this problem, we chose to work on the result matrix, in which the polygons are represented by strings of the symbols S, *, +, E and I with one pixel width. This procedure is equivalent to converting the results from the raster representation to a connected vector format. In the process of coincident line extraction we use the operation for three different purposes. In the first two cases we compile only the newly extracted and not yet verified polygons (strings of symbols S, * and +) and store them in the files MLPIICL respectively MLPIRCR. Simultaneously the strings of symbols are erased, because only the lines accepted shall be reinserted to be retained after the verification process. In the third case we compile the list of final results (strings of symbols E and I) and store them in the files FMLPIICL respectively FMLPIRCR. In general, the algorithm starts at the dead ends, which mark the ends of the polygons in question.

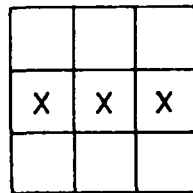
The maximum length polygon compilation follows the polygons in the result matrix by evaluating a 3 x 3 window. Figures 4.2a - h show examples of symbol distributions. The center of the matrices marks the current position of the procedure. In figure 4.2a and b, a dead end situation is displayed because symbols radiate only to one direction. Figures 4.2c - e show normal situations during the line following, because two of the neighbouring elements of the center have an entry. In figures 4.2f - h an intersection may have been detected, because the surrounding elements of the center show more than two entries.



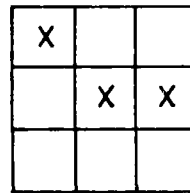
a)



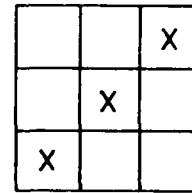
b)



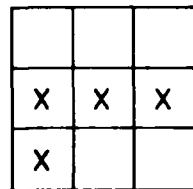
c)



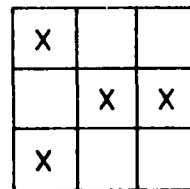
d)



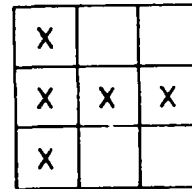
e)



f)



g)



h)

Fig. 4.2: Examples of situations during polygon following

If a situation as displayed in figures 4.2f - h is present, a crossing examination is performed to detect the separate branches, which constitute the intersection. By evaluating a 5 x 5 window with the same central point as the 3 x 3 window the proper directions are calculated (see figure 4.3). In some cases it is recognized that the algorithm is not at the position of an intersection but at a sharp bend of the polygon (figure 4.3c). In figure 4.3a and d situations are shown, where an intersection was found, but the center point of the window under consideration is not identical to the middle point of the crossing. Because we know the correct location of the middle point of the crossing, we have detected the end of a maximum length polygon. Starting from the intersection the algorithm follows the string of symbols of the other branches.

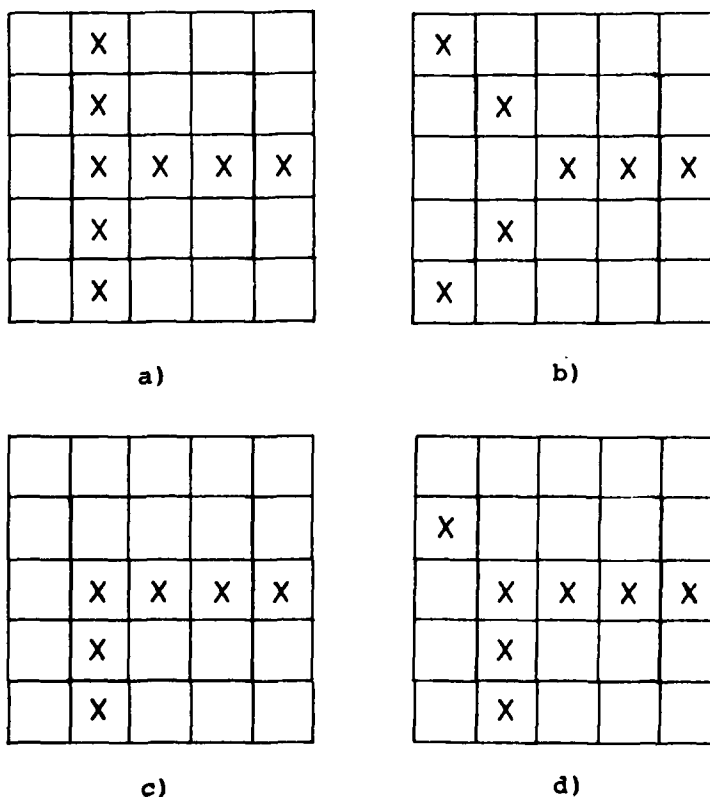


Fig. 4.3: Examples of crossing examinations

In the process of coincident extraction of line objects from stereo image pairs three modifications of the maximum length polygon compilation method are used. For explanation purpose we will assume that we want to compile polygons in the left image only:

- when operating on the initial extraction results the compilation starts from the break off dead ends, which have been created during the initial extraction process and listed in the file BDEIILCL; all polygons found are extracted and erased from the result matrix
- when operating on continuation extraction results the maximum length polygon compilation starts from the verification dead ends (as defined in section 4.4.1). Only the symbols S, * and + are extracted and erased, because the other entries I and E of the result matrix mark polygons already accepted. Let figure 4.4a show a part of the result matrix before the continuation extraction. If the dashed line AB of figure 4.4b marks the newly detected object part starting from two opposite verification dead ends, the line will not be detected without using the verification dead ends. During the first continuation extraction, starting points are searched, because it is the first extraction in the stereo partner image and objects may be detected, which could not be detected in the other image. Analogous to the initial compilation we have to use the break off dead ends of these line parts

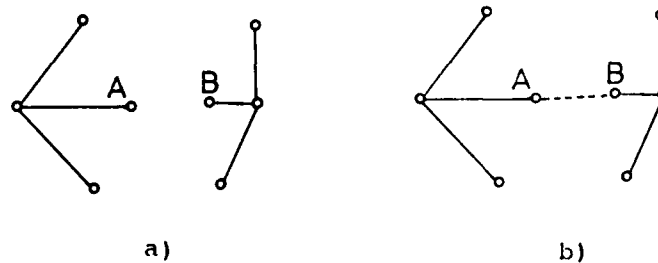


Fig. 4.4: Example of a continuation extraction

- when compiling the list of final results the maximum length polygon compilation starts from the absolute final dead ends. While in the two other cases the symbols S, + and * were extracted, the symbols E and I are extracted now. Furtheron no erasing takes place, for the result matrix is kept as graphical final result.

4.3 Image registration algorithms

The process of coincident line extraction from stereo image pairs requires a registration function between the two images. This registration shall map coordinates from one image into corresponding coordinates of the other image. The inverse mapping of coordinates from the second into corresponding coordinates of the first image will also be required (fig. 4.5). We want to use this function to map the gray level representation of an image as well as extraction results (polygons) of an image. In the system under discussion we actually apply it to extracted polygons of an image. The performance of our extraction methods does not require a perfect registration, i.e. misregistration of up to 12 pixels can be tolerated.

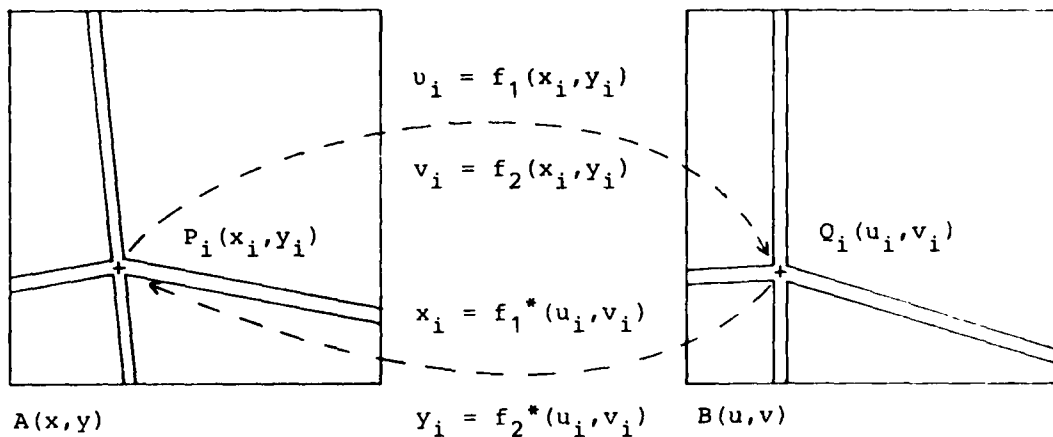


Fig. 4.5: Mapping of a point $P_i(x_i, y_i)$ of the coordinate system $A(x, y)$ into a corresponding point $Q_i(u_i, v_i)$ of the coordinate system $B(u, v)$ and vice versa

For the purpose of registration we have investigated two different algorithms. In what follows we will describe how they work in the case of mapping coordinates of an image $A(x,y)$ into coordinates of an image $B(u,v)$.

The main steps of both algorithms are the following:

- selection of pairs of control points (corresponding locations) $P_k(x_k, y_k)$ in $A(x,y)$ and $Q_k(u_k, v_k)$ in $B(u,v)$
- determination of the coefficients of the mapping functions

$$u = f_1(x, y) \quad (4 - 1)$$

$$v = f_2(x, y) \quad (4 - 2)$$

and of the inverse mapping functions

$$x = f_1^*(u, v) \quad (4 - 3)$$

$$y = f_2^*(u, v) \quad (4 - 4)$$

by means of the selected pairs of control points

- performance of the mapping process.

The selection of pairs of corresponding control points is partially automated. In general we will select corresponding features in both images visually and mark their respective location on our COMTAL display. Particularly well suited locations for the selection of control points are locations with high contrasts and for our purposes such locations on roads or in the neighbourhood of roads like crossings of roads or buildings near roads (fig. 4.6).

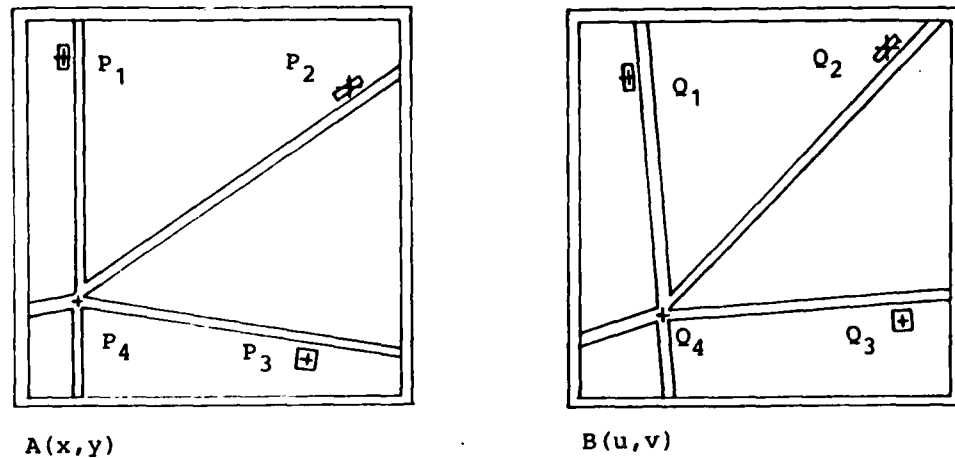


Fig. 4.6: Visual selection of corresponding control points on a COMTAL display

If necessary we can achieve a higher accuracy of the coordinates of the control points by identifying them in printer outputs of the respective image segments. For example we can identify the center of gravity of an imaged building with more confidence (fig. 4.7).

The main difference between both registration algorithms is their use of different mathematical models:

- a projective mapping,
- a polynomial mapping.

168	177	178	175	193	186	179	182	176	176	175	176	185	182	187	167
162	173	166	157	160	161	151	158	173	202	194	185	179	187	180	165
127	130	130	130	146	149	143	145	199	214	221	220	208	191	185	175
129	121	128	176	206	218	225	216	231	231	233	231	225	191	185	184
114	111	121	146	179	219	223	226	229	224	224	222	223	200	183	174
123	125	142	164	188	229	238	244	241	241	229	224	212	196	185	182
134	125	138	175	217	229	241	245	241	230	208	192	194	179	191	184
135	131	142	206	236	238	234	240	231	230	214	193	188	178	183	177
163	150	172	214	231	236	240	227	233	231	215	178	186	196	201	201
180	161	193	203	216	227	217	225	230	220	202	179	179	191	193	185
170	182	179	177	169	173	197	187	178	160	189	177	174	168	183	179
167	172	170	160	158	164	160	163	161	161	174	173	170	174	169	176
149	185	190	177	160	151	195	181	185	175	164	145	160	147	163	160
166	188	159	166	163	157	172	166	189	187	183	168	141	163	169	148
150	169	165	162	192	200	204	209	200	220	222	204	191	162	166	130
116	147	130	188	200	232	235	247	244	254	247	236	224	165	155	144
108	103	118	166	165	220	235	248	255	252	249	244	223	181	174	149
124	110	107	158	194	238	250	250	255	250	244	236	218	185	163	166
106	98	124	187	229	248	246	246	251	240	228	207	196	155	147	145
115	99	128	198	234	241	251	247	245	238	212	174	170	166	157	146
100	115	157	226	238	246	254	243	247	224	183	155	158	161	143	177
109	115	125	204	227	241	236	241	242	225	209	181	165	169	173	156
138	149	138	186	192	202	213	204	226	212	177	168	159	163	159	175
164	160	176	161	161	143	183	172	171	164	143	150	158	171	181	163
166	158	138	147	135	130	149	170	149	160	156	162	147	177	154	156

Fig. 4.7: Printer output of corresponding locations in two images

The projective mapping function /4/

$$u = \frac{a_0 + a_1 * x + a_2 * y}{c_0 + c_1 * x + c_2 * y} \quad (4 - 5)$$

$$v = \frac{b_0 + b_1 * x + b_2 * y}{c_0 + c_1 * x + c_2 * y} \quad (4 - 6)$$

provides a strictly affine mapping (translation, rotation, scaling, projective distortion). The equations contain nine unknown coefficients a_1 , b_1 and c_1 which reduce to eight after division by one nonzero coefficient. It is possible to set up a linear equation system which allows the determination of the eight coefficients by means of four pairs of control points. To guarantee the greatest possible accuracy the control points should be located in the corners of the overlapping area of both images.

Initially we wanted to apply the projective mapping. From the beginning, we were aware of the fact that exact matching by this method can only be achieved under three conditions:

- the selected control points must be properly distributed
- the selected control points must lie on one plane
- only points of this same plane can be mapped correctly with respect to our requirements.

And we were also aware of the fact that it is usually rather difficult if not impossible to satisfy all three conditions. So we expected a certain amount of deviations of the mapped lines from the actual lines. But after the test of this method we realized that the differences were bigger than we had expected, possibly due to considerable elevations of the terrain imaged. So we had to look for a different method. We finally settled with the polynomial mapping, which proved to be well

suited for our purposes. The polynomial mapping function /5/ has the following form:

$$u = g_0(x) + g_1(x, y) \quad (4 - 7)$$

$$v = h_0(y) + h_1(x, y) \quad (4 - 8)$$

The mapping function consists of a part (g_0 and h_0), whose task it is to take care of uniform, global distortions and another part (g_1 and h_1), whose task it is to eliminate the local differences. In the images, which we have processed, the only uniform distortion is a translation in x- and y-direction. So the functions g_0 and h_0 (4 - 9 and 4 - 10) are defined accordingly. For this purpose the algorithm requires one pair of control points $P_0(x_0, y_0)$, $Q_0(u_0, v_0)$. The separate elimination of global distortions between both images by g_0 and h_0 is optional, i.e. we need not compute the two functions g_0 and h_0 . The functions g_1 and h_1 (4 - 11 and 4 - 12) will do the job, too. However, if we do compute g_0 and h_0 we will achieve a satisfactory mapping more easily.

$$g_0(x) = x - (x_0 - u_0) \quad (4 - 9)$$

$$h_0(y) = y - (y_0 - v_0) \quad (4 - 10)$$

$$g_1(x, y) = \sum_{k=1}^n a_k * \text{SQRT}[(x-x_k)^2 + (y-y_k)^2 + SF] \quad (4 - 11)$$

$$h_1(x, y) = \sum_{k=1}^n b_k * \text{SQRT}[(x-x_k)^2 + (y-y_k)^2 + SF] \quad (4 - 12)$$

n: number of control points

For the determination of the n unknown coefficients a_k and b_k it is necessary to solve the following linear equation systems:

$$\bigwedge_{1 \leq i \leq n} \left[\sum_{k=1}^n a_k * \text{SQRT}((x-x_k)^2 + (y-y_k)^2 + SF) = u_1 - g_0(x_1) \right]$$

(4 - 13)

$$\bigwedge_{1 \leq i \leq n} \left[\sum_{k=1}^n b_k * \text{SQRT}((x-x_k)^2 + (y-y_k)^2 + SF) = v_1 - h_0(y_1) \right]$$

(4 - 14)

with n control points $P_j(x_j, y_j)$ in A(x,y)

n control points $Q_j(u_j, v_j)$ in B(u,v)

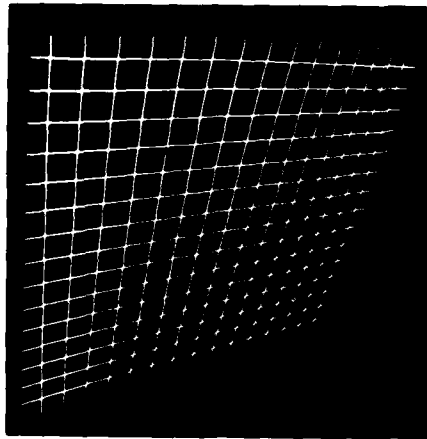
$$\text{smoothing factor } SF = c * \min_{\substack{i, k=1, n \\ i \neq k}} \left[(x-x_k)^2 + (y-y_k)^2 \right]$$

(4 - 15)

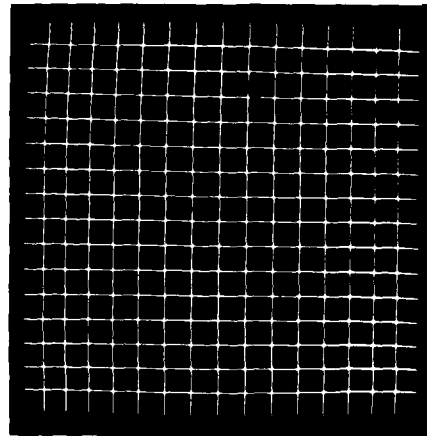
constant c

number of control points n.

The number of linear equations is always equal to the number of unknown coefficients and both are equal to the number of control points. In the case of the polynomial mapping it is advisable to locate some control points in the corners of the overlapping area of both images as well as at those parts of the images which are of special interest. In both cases, in the case of the projective mapping as well as in the case of the polynomial mapping two sets of coefficients are computed, one for the mapping from A to B and the inverse one for the mapping from B to A. The character of both methods can be illustrated very well by the rectification of a grid (fig. 4.8, fig. 4.9).

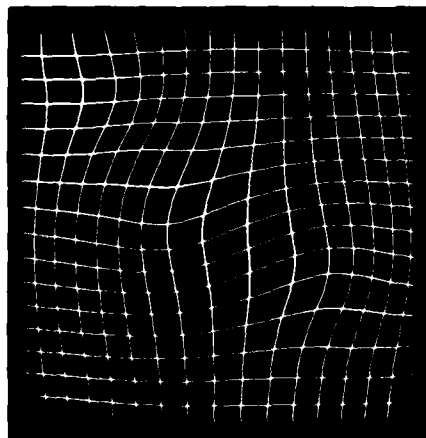


a)

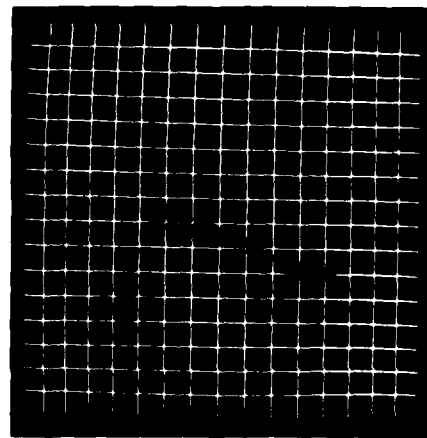


b)

Fig. 4.8: Registration with the projective method
a) distorted grid; b) rectified grid



a)



b)

Fig. 4.9: Registration with the polynomial method
a) distorted grid; b) rectified grid

We will close the description of the registration algorithms with some details concerning the polynomial registration method:

A registration with this method is in essence an interpolation between control points. The smoothing factor SF causes the interpolated plane to be more or less smooth. With $c = 0.6$ for example it will be smoother than with $c = 0$.

Theoretically the number of pairs of control points has no upper limit. In the case of the $8000 * 8000$ pixel images that we have processed we have experimented with 10 to 65 pairs of control points and have achieved good to excellent results using sets of 15 or more pairs of control points. Finally, it took approximately 5 seconds CPU time at the different stages of the process for mapping the extraction results from one image into the other image.

4.4 Verification procedure

The verification procedure is used to test the prediction that a line object with certain predefined features exists at a given location in the image. For instance if we have extracted some object parts in the left image of a stereo pair, we want to examine, whether we can find the same object parts also in the right image and vice versa. Only if the line object can be extracted in both images at corresponding locations it is accepted as final result. This tends to be very restrictive, because even if the line is clearly visible in one image the corresponding part in the stereo partner may be occluded and therefore the result is discarded.

We have to map the maximum length polygons, which are listed in the file MLPILCL created by the maximum length polygon compilation, from the coordinate system of IL to the coordinate system of IR to get the corresponding locations. This will be

done with the polynomial registration algorithm described in section 4.3. We distribute the control points in the overlapping area of the stereo pair to guarantee that the predicted locations for line objects differ only to a small amount from the correct locations. A maximum of 12 pixels misregistration can be tolerated. The mapped maximum length polygons extracted in IL (listed in the file MLPILCR) are used by the verification procedure as predictions for the existence of line objects in IR. At the predicted locations in IR the verification procedure tries to find a line object of the predefined kind by use of a modification of the regional extraction method, which is shortly described in section 4.1 and in more detail in /1/. If parts of the line object can be found the respective parts of the polygons are accepted as final results for both images. They will be entered in the lists of accepted polygons APLIRCR and APLILCR. The latter one has to be mapped back to the coordinate system of IL to produce APLILCL. To get a final graphic result they are also entered in the result matrices of both images, in RMIR with the symbol E for external and in RMIL with the symbol I for internal. If it is not possible to extract the line objects in the right image at the predicted locations, the extraction results from IL may possibly be wrong and are discarded. Starting from the dead ends, which are defined by the end pieces of accepted polygon parts, a continuation extraction can take place.

In the following sections the different parts of the verification procedure are described in more detail.

4.4.1 Kernel of the verification procedure

The mapped extraction results are listed in the file MLPILCR respectively MLPIRCL as polygons of maximum length, which means that each polygon ends either in a dead end or in an intersection. Each of the polygons is defined by the set of its vertices. According to the direction of the segments a chain of areas of interest (AoIs) is positioned along each polygon (see fig. 4.10).

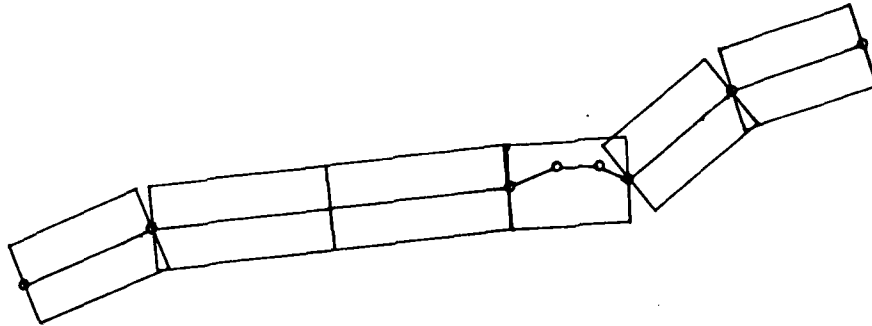


Fig. 4.10: Example of an AoI chain

Let d be the distance between two neighbouring vertices P , Q and d_m , d_M the minimal and maximal allowed lengths of an AoI. If formula

$$d_m \leq d \leq d_M \quad (4 - 16)$$

is valid, then the two vertices P , Q are taken as end points of an area of interest. An AoI should have a minimal length to guarantee that a part of the line object is contained in the AoI, which is long enough to allow a profound decision on its properties. Theoretically no upper limit exists for the length of an AoI, but within the implementation the length must not exceed a practical limit.

If the distance between two neighbouring vertices P and Q is less than the minimal required length d_m , it is examined, if the following segment QR of the polygon can be combined with the segment PQ to define an AoI. This is possible, if the angle between PQ and QR does not exceed an angle ALPHA , where ALPHA

limits the maximal allowed curvature of a line object in an AoI. If the length of the resultant PR is still less than the minimal required length of an AoI, it is examined, if the following segment RS can also be used to define a common AoI. This is possible, if the angle between QR and RS and the angle between PQ and RS is less than ALPHA.

If the angle between such short neighbouring segments is greater than ALPHA an AoI must be defined for each segment separately. With the chosen degree of resolution only very few (if any) curves occur, which have such a high curvature. If the length of a polygon segment PQ is larger than the maximal allowed distance dM, the segment PQ is equidistantly divided until the parts fulfill formula (4 - 16). The width of the AoI is chosen big enough to cope with the amount of misregistration of the mapping algorithm.

Each maximum length polygon of the mapped results is examined separately. At this stage of the procedure no difference is made whether a polygon ends in a dead end or an intersection. The branches of a node of the line network will be identified during a reconstruction step for the crossings (see section 4.4.3). Each area of interest of a chain is examined with the regional extraction method according to the predefined description of the line object searched.

After the evaluation of two neighbouring AoIs of a chain the distance between the two endpoints of the object parts accepted (if any), which point to each other is calculated. If the distance is very small (less than 10 pixels) the two verification results are linked without further examination. If the distance is of medium size (say 10 to 20 pixels) a comparison of the directions of the two found parts is done. If the parts are nearly collinear (if they differ in the direction to a small amount) they are also linked. Otherwise an intermediate AoI is defined by the two endpoints directed to each other

(see fig. 4.11). Let ABC and DE be two object parts, which could be detected by evaluating two neighbouring AoIs, then an intermediate AoI with endpoints C and D is defined, because the angle BETA between AC and DE does not permit, to interpolate the line CD. If a new part of the line object can be extracted, it is connected to the two previously found parts. An intermediate AoI will also be defined if the distance d_i between the object parts is greater than 20 pixels but less than d_M . If a bigger gap has been produced the area will not be searched any more and a final dead end is defined at the open ends of the polygon parts. An intermediate AoI is also examined, if for the distance d_f between the first found point of the first AoI of a chain and the first point of the mapped polygon formula (4 - 17) is valid. The same is done for the distance d_l at the other end of the polygon.

$$20 < \begin{pmatrix} d_i \\ d_f \\ d_l \end{pmatrix} \leq d_M \quad (4 - 17)$$

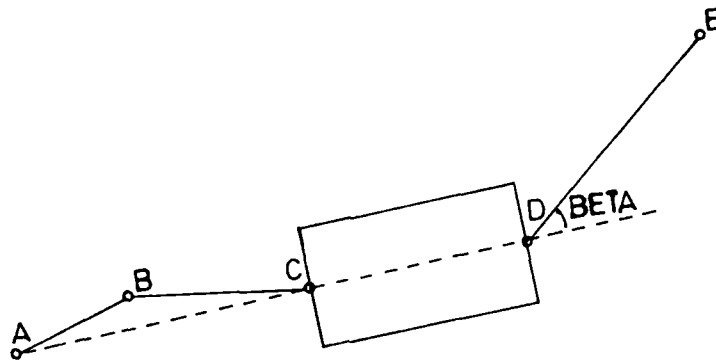


Fig. 4.11: Example of an intermediate AoI

If a gap in the verification result cannot be bridged by evaluating an intermediate AoI we want to accept only these parts of the mapped extraction line, which correspond to the parts of the line object, which we have found. Therefore the endpoints of the accepted polygon parts are projected onto the mapped maximum length polygon and the remainder of the line is cutted off. Let R' , S' be the endpoints of the accepted polygon then RS will be the accepted part of the verification input, see figure 4.12.

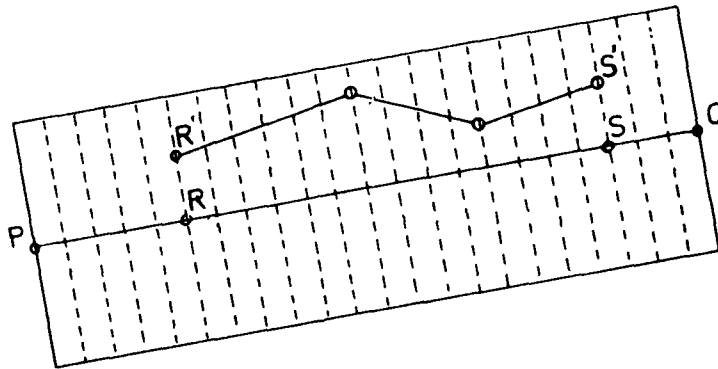


Fig. 4.12: Example of the projection of an accepted polygon part onto the verification input

There are three different kinds of acceptance for a polygon:

- a polygon is completely accepted
- a polygon is partly accepted
- a polygon is not at all accepted.

The accepted polygon parts are listed in the file APLILCL respectively APLIRCR and the accepted input polygon parts in the file APLIRCL respectively APLILCR. These have to be mapped back to the image, in which they were extracted originally and

inserted in the result matrix. This will be explained in more detail in section 4.4.2. The accepted polygon parts will be entered in the result matrix of the image, in which they were verified, with the symbol E (external). This signifies that they were extracted originally elsewhere (in the image of the stereo partner).

To get an access to the separate parts of the accepted polygons two different kinds of dead ends are defined at their ends:

- verification dead ends:

These will be used for a continuation extraction and defined at the ends of the accepted parts of a polygon nearest to the endpoints of the verification input. Let AB be a mapped maximal length polygon, $P'Q'$, $R'S'$ and $T'U'$ three accepted parts and $P''Q''$, $R''S''$ and $T''U''$ their projections. Then a verification dead end will be defined at the endpoints P' and U' , if the line AP'' respectively $U''B$ is only a few pixels long (see figure 4.13). The verification dead ends will also be used for the reconstruction of intersections.

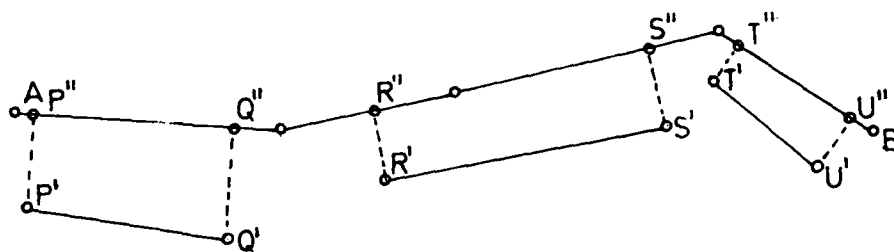


Fig. 4.13: Examples of verification dead ends

- final dead ends:

These will be used during the final result processing. No further extraction is tried at these locations in any of the images.

4.4.2 Backmapping and reinsertion of accepted polygons

After all mapped maximum length polygons have been processed by the verification algorithm, some accepted polygon parts have probably been produced. To get a final result for both images containing all accepted polygon parts, independent of the image in which they were extracted and in which they were verified, with corresponding coordinates respective to the correct coordinate systems, the accepted parts of the verification input have to be mapped back to the coordinate system of the image, in which they have been extracted originally. The result of the backmapping will be listed in file APLILCL respectively APLIRCR. The lines contained in these files will be entered in the proper result matrix with the symbol I, marking that these lines were extracted from the related image and verified in the stereo partner image.

During the extraction process all lines found were inserted in the result matrix, but erased during the maximum length polygon compilation. By reinserting the accepted parts we can distinguish results already verified from results newly extracted (not yet verified). Furtheron it is possible to distinguish during the final result processing, where a certain line was originally extracted and in which image it was verified.

Due to accessibility and storage reasons the backmapping can take place immediately after the verification processing but not the reinsertion of the accepted polygon parts. During the verification processing segments of the image gray level matrix and the result matrix must be accessible. This is only possible according to our computer system installation, if both

files are loaded via a disk. The disk has only space enough for two files of such a big size, for we need about 125000 blocks of 512 bytes each out of 340000 for one file. If we would like to reinsert the accepted polygons during the run of the verification program, we had to use a second disk for the result matrix of the stereo partner, which was not available.

Therefore the reinsertion of the accepted polygons is done before the results of the continuation extraction from the stereo partner image are submitted to verification. Hence before the final result processing the reinsertion of the last verified and accepted results has to be done.

4.4.3 Reconstruction of intersections

For explanation, it is assumed that the line network was extracted in IL and verified in IR. In the verification process gaps may occur at positions in IR, where a polygon of IL was only partly accepted. Often such gaps occur at intersections of polygons. A typical example for such a gap is shown in figure 4.14. In a further treatment of the line network a reconstruction of such intersections is possible.

The coordinates of the vertices of polygons accepted are stored in the file APLIRCR. The verification dead ends (see section 4.4.1) are stored in the file VDEIRCR. Segment PQ (fig. 4.14) is an example for such a dead end. During the process of reconstruction, these files are updated, and the results are entered into the result matrix of IR: The reconstruction of intersections is achieved by searching a segment QC inside a certain area, with minimum distance between the dead end PQ and another polygon. For this purpose an area of investigation, which has the shape of a kite, is defined. If a segment QC can be found inside the area of investigation, the symbol E (external) will be entered into the result matrix and the coordi-

nates of the vertices Q and C of the new polygon segment are stored in the file APLIRCR, and the dead end PQ is deleted in the file VDEIRCR.

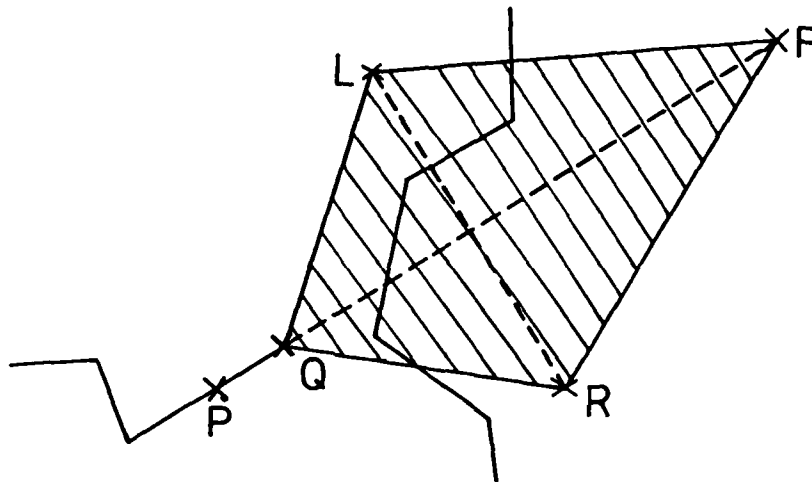


Fig. 4.14: Example of an incomplete intersection with the defined area of investigation

The reconstruction of intersections takes place in the result matrix. The investigation proceeds as follows:

1. The border of the area of investigation is entered into the result matrix with the symbol B as an identifier. The identifier is only entered, if no symbol is already entered in the matrix element of the border. The starting point Q of the area of investigation is identical to the second point of the dead end PQ. The main axis QF is defined in the same direction as the dead end. Our tests of the images available have shown that the lengths of the axes should be set to 20 matrix elements for the main axis QF and 12 matrix elements for the smaller axis RL.

2. For the search inside the kite, the matrix elements of the result matrix in the area are investigated in a special sequence of elementary steps. Such a sequence results from a step in x- and y-direction, both are not greater than one matrix element. The sequence of the steps is according to the numbers in figure 4.15. The first step is done on the main axis QF, starting with the point Q. If there is no entry in the result matrix at this position, the search is continued on "circles" with regard to the central point Q. The first entry of the result matrix which can be found inside the area of investigation is the matrix element C (see fig. 4.16), which has the minimum distance to the starting point Q. In this search algorithm the direction of the main axis is favoured.

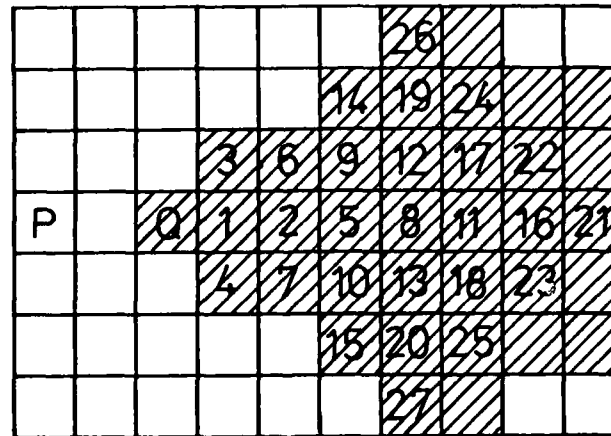


Fig. 4.15: Diagram of the search algorithm

3. In a further part of the procedure the segment QC will be interpolated. Hence the symbol E (external) is entered into the result matrix. This entry signifies that these matrix elements are entered in the result matrix of the image, in

which they are verified. In the final result processing (see section 4.5), in which this algorithm is used, too, matrix elements of such segments are entered with the symbol I (internal). In this case "internal" means that symbols are entered in the result matrix of the respective image. Additionally, the coordinates of the vertices Q and C of the segment are stored in the file APLIRCR and dead end PQ is deleted in the file VDEIRCR. Fig. 4.16 shows the result of an investigation.

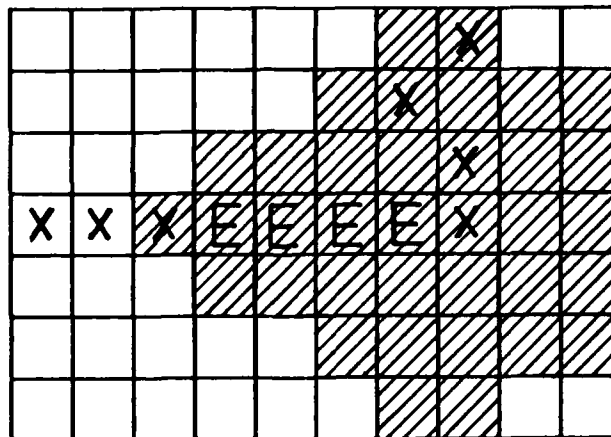


Fig. 4.16: Principle sketch for the interpolation of the segment QC in an area of investigation

Additionally to the reconstruction of intersections, it is possible to complete the line network at other positions, too. Polygons, which are interrupted by short gaps can be completed with the help of this algorithm (see section 4.5).

4.5 Final result processing

The final result processing is the last procedure of the coincident line extraction from stereo image pairs. It is used to compile the lists of final result polygons of both images related to their respective coordinate systems. Thus we get two lists of polygons representing corresponding line objects, which can be used to calculate the parallaxes in the overlapping area of the two images.

After a few iterations of extraction and verification, the next continuation extraction process will produce only a small amount of new results. If this does not contribute significantly to the total result accepted up to this moment it is not worthwhile to continue the cycle of extraction and verification processing.

At this moment there exist several files concerning the final results:

- the result matrices RMIL and RMIR containing all accepted polygons in raster representation
- the files APLILCL and APLIRCR with the lists of accepted polygons
- the files FDEILCL and FDEIRCR containing the final dead ends.

Due to transportability and storage reasons it is much more desirable to save the final result polygons in the vector format, but the polygons of APLILCL and APLIRCR are stored on a first come first store basis, even so the result of each extraction and verification step is locally sorted. Because this is very unsuitable for further processing, the list of final maximum length polygons is compiled for both images. Fig. 4.17 shows the sequence of processing steps to get the list of maximum length polygons for IL.

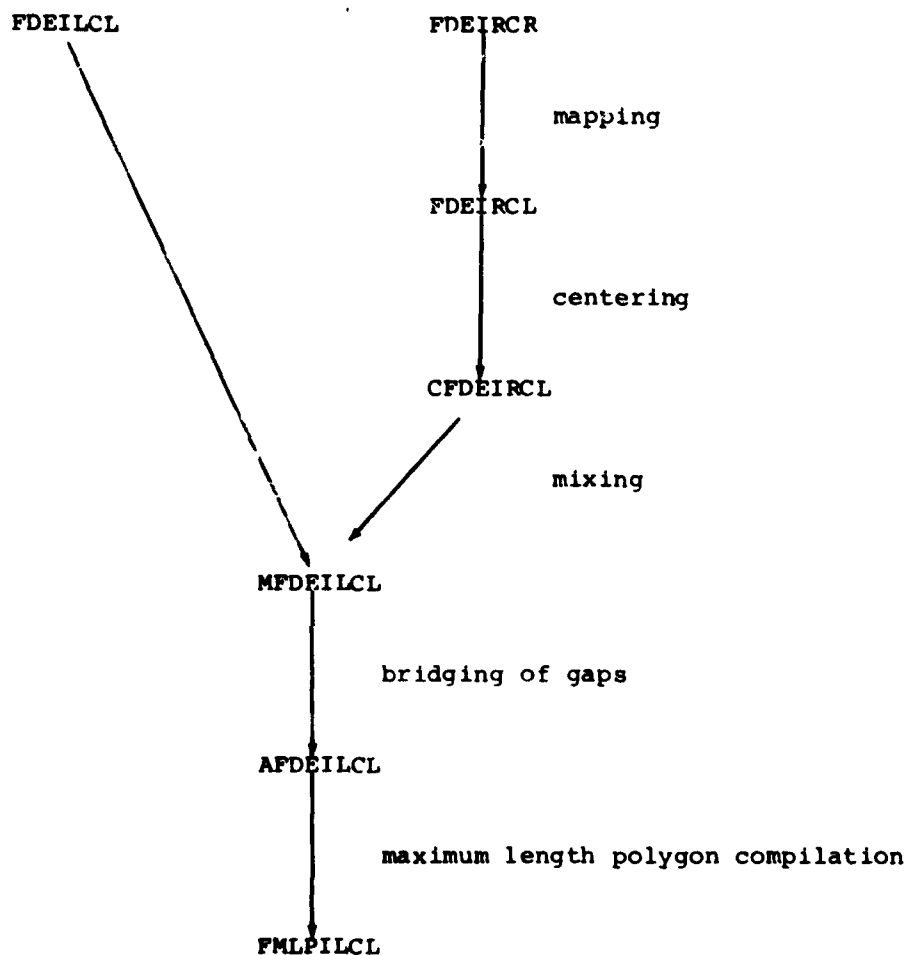


Fig. 4.17: Sequence of the final result processing steps

Starting from the final dead ends listed in the files FDEILCL and FDEIRCR the maximum length polygons are calculated. The final dead ends are defined in six different cases listed below:

- at the ends of a partly accepted polygon, if the first respectively last accepted point is more than a few pixels distant from the end of the verification input (see section 4.4.1)

- at the ends of each accepted polygon part (except these mentioned above) if a polygon could only partly be accepted
- at the ends of an earlier accepted result, if the first accepted point of the appending mapped polygon is more than two pixels apart
- at the ends of an earlier accepted result, if the appending mapped polygon could not at all be accepted
- at the ends of an accepted polygon, if no continuation extraction was possible at this location
- at the ends of an accepted polygon, if a continuation extraction was possible but it was decided that the cycle of extraction and verification should be stopped and no verification processing was done for the newly detected polygon.

Notice that the sets of final dead ends are totally disjunct, for if we define a final dead end, we have already examined the corresponding location in the other image of the stereo pair and either defined at this position a break off dead end or we found a part of a line object. Therefore if we want to calculate the final result with the maximum length polygon compilation for example IL, we need the dead ends of the file FDEILCL as well as these of the file FDEIRCR and we have to map the dead ends listed in FDEIRCR from the right to the left image coordinate system. This will yield the file FDEIRCL.

Due to the slight inaccuracy of the (polynomial) registration algorithm we get some deviations between the mapped final dead ends and the ends of the accepted polygons in RMIL. By using another modification of the regional extraction method REM we cope with these irregularities and shift the mapped final dead ends to the correct locations. This will be called centering and works as follows:

Let P, Q be the endpoints of a mapped final dead end. The line PQ has approximately the correct location and direction. To center the dead end represented by the line PQ we choose an area of interest for evaluating by REM with the same direction

as the dead end. We elongate the line PQ to both sides to get a line P'Q' with a length of about 40 pixels (see fig. 4.18), because a dead end is very short (about 6 pixels). The points P', Q' define the endpoints of the AoI. The width of the AoI is chosen equivalent to the width of the AoIs used in the verification process. By choosing this area of interest it is guaranteed that we can detect the end of the accepted polygon. This is very easy because the result matrix does not present the difficulties of a gray level matrix, it contains only entries with the symbols I and E. Hence we need not make a profound examination of the AoI. The centered mapped final dead ends are listed in the file CFDEIRCL.

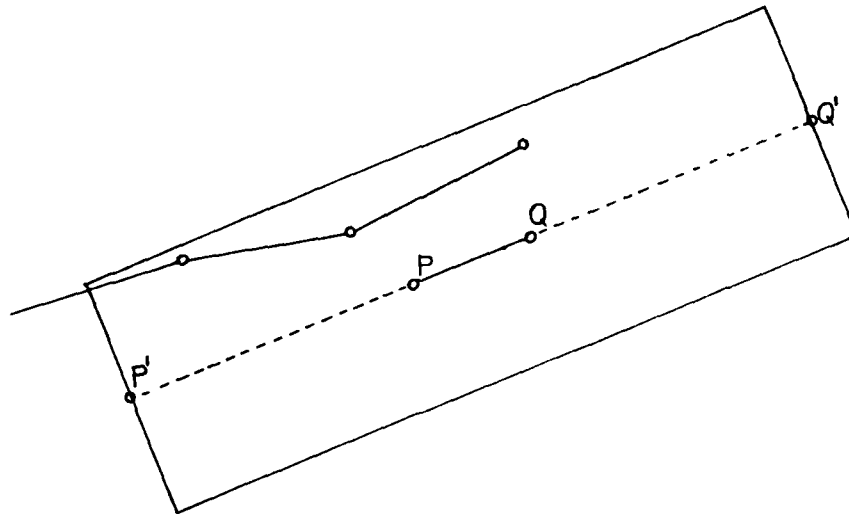


Fig. 4.18: Example of the centering of a final dead end

Now the files FDEILCL and CFDEIRCL can be mixed and sorted according to increasing values of coordinates to reduce the loading time of result matrix segments during execution of the next steps. This will yield the file MFDEILCL.

For each dead end listed in MFDEILCL it is examined, if a polygon is placed in the neighbourhood. There exist two main possibilities that a dead end is encountered, which points to another polygon nearby (see fig. 4.19). If the polygon on the left hand side of figure 4.19a is accepted at an earlier stage of the process cycle than the vertical line, we have extracted an intersection, which was not recognized as such till now. In figure 4.19b we have another situation. Let again the polygon on the left hand side be accepted at an earlier stage. If we had got another extraction result coming from the right hand side up to the end of the accepted line and the last few pixels of this result could not be accepted, we get a small gap in both images. But only the gap in the result matrix of the image, in which the verification of the later result was executed, was bridged as described in section 4.4.3. This algorithm is also used now to bridge any small gap left. If a connection with another polygon is possible the respective dead end will be deleted, so we get the file AFDEILCL with the absolute final dead ends.

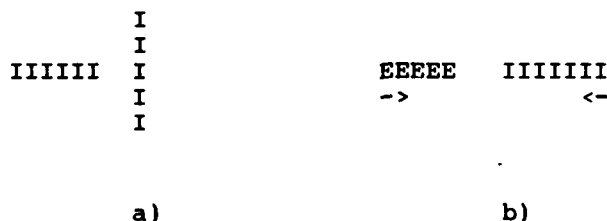


Fig. 4.19: Examples of small gaps

Starting from the dead ends listed in AFDEILCL we execute the maximum length polygon compilation to get the final result list FMLPILCL for IL.

To get the list of maximum length polygons also for IR, we have to perform the final result processing once more, this time on the basis of RMIR and by mapping the file FDEILCL to the right image coordinate system.

5. Presentation of the results

5.1 FIM image processing system

The FIM image processing system configuration is shown in fig. 5.1. The transparencies delivered were scanned on our OPTRONICS C4500 device. Each image was scanned with a 25 μ m raster to produce image matrices of 9201 x 9201 pixels stored on magnetic tape (three tapes per image). To get rid of some distorted parts at the image boundaries and to comply with the idea of mosaicing the image into overlapping segments of 1024 x 1024 pixels each, we selected a central part of 7968 x 7968 pixels of each image scanned as input data to our extraction procedure. For both stereo image pairs the overlapping area amounts to a size of approx. 4200 x 7968 pixels. When stored on a RP 06 disk as input data to the extraction procedure, each image occupies approx. 64 Mbyte of the 176 Mbyte storage available. As the procedure requires a result matrix of the same size for each image, it was not possible to provide both images of the stereo pair for direct access at the same time. Therefore, we decided to produce separate extraction results from each of the big images of 7968 x 7968 pixels each and to submit sections of 4000 x 4000 pixels to the process of coincident line extraction. These sections of the stereo image pairs covered the interesting parts of the overlapping areas and could be stored on disk simultaneously. There is a third reason for this decision pertaining to the computational time required for the complete process; this will be explained in the following.

5.2 Extraction results in images of 8000 x 8000 pixels

To demonstrate, analyze and assess the performance of the extraction procedure we applied the process to all three images separately, using different parameter sets. For this sequence of results, three variable parameters will demonstrate the

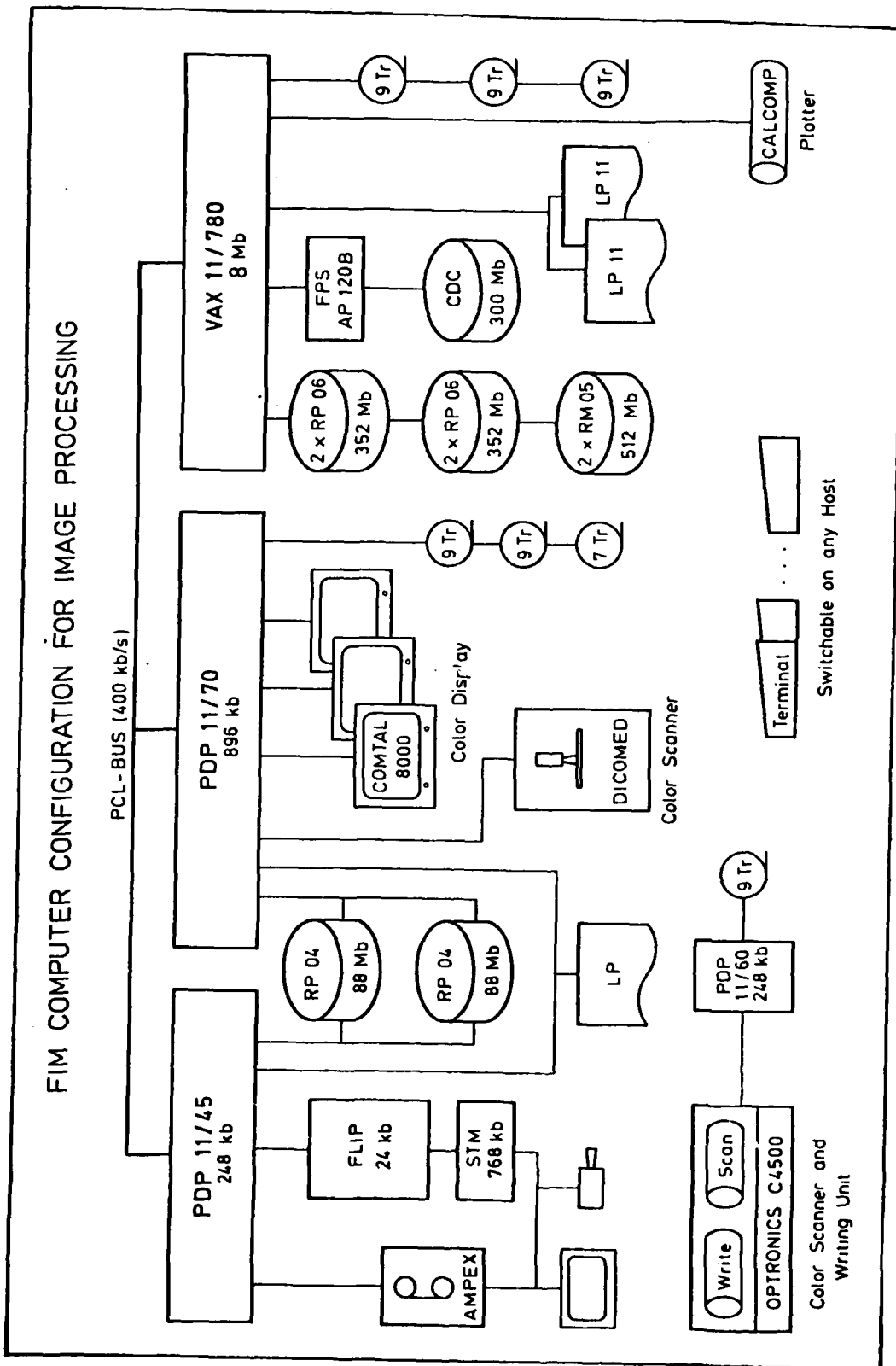


Fig. 5.1: FIM image processing system configuration

flexibility, the robustness and the performance of the extraction methods:

- the different input data, representing the same area of the ground
- the different width that we allowed for the line objects
- the different amount of contrast that we required for a peak to be accepted as cross section of a line object. The type of contrast was always set to "bright", i.e. we were always extracting roads, only.

The following figures show prints of the three aerial images of 7968 x 7968 pixels which we used:

- transparency No. 52 has been renamed S5105S and is shown in fig. 5.2
- transparency No. 54 has been renamed S5107S and is shown in fig. 5.3
- transparency No. 56 has been renamed S5109S and is shown in fig. 5.4.

On each figure a transparent grid is overlayed to indicate the location of 8 x 8 subimages which will be referenced during the explanation of the results. The first series of extraction runs was performed on each input image using the standard parameter value set, including

- amount of contrast required: 15 gray levels
- width of the line objects: 2 to 5 pixels

The results are shown in figures 5.5 through 5.7 for the images No. 52, No. 54 and No. 56 respectively. Regarding these results a few general remarks are obvious:

- the characteristics of the results are unchanged when compared to the previous project /1/; the network of roads is not totally complete and not always connected; there are some gaps, but no significant faults

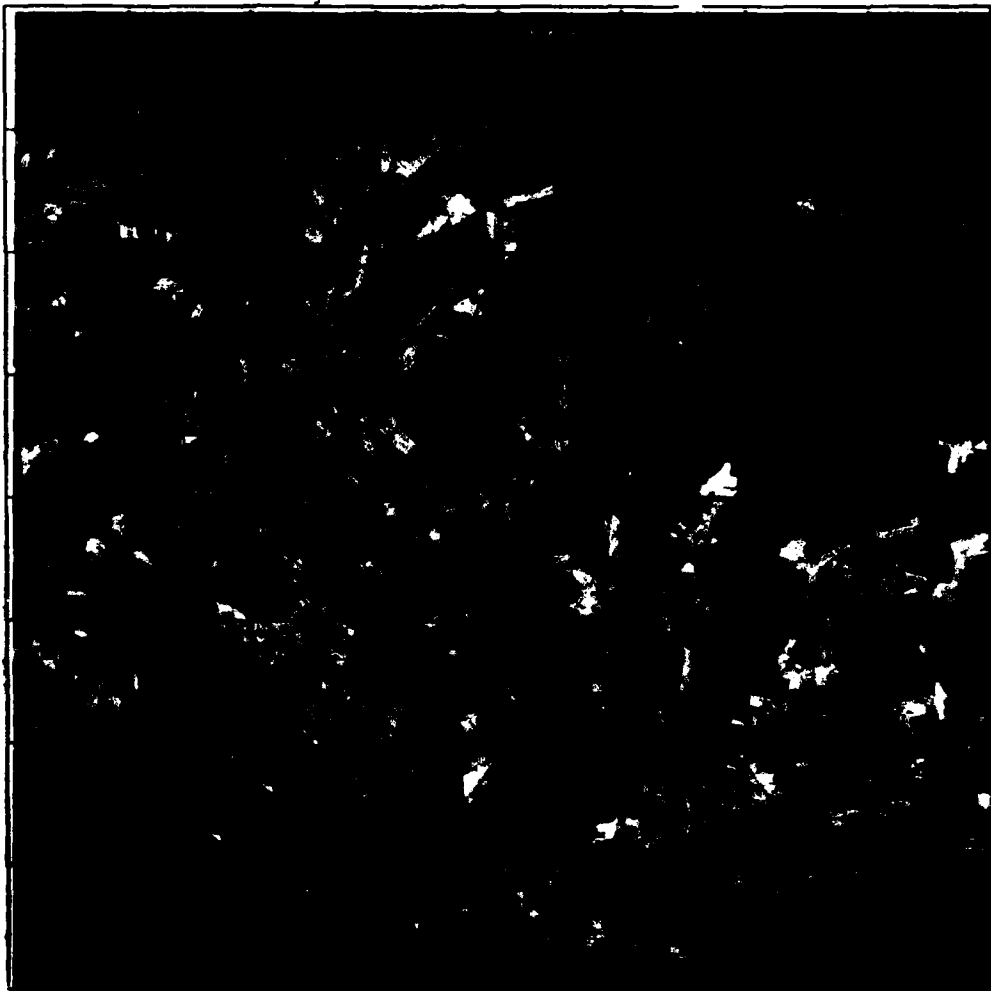


Fig. .2: Transparency No. 52

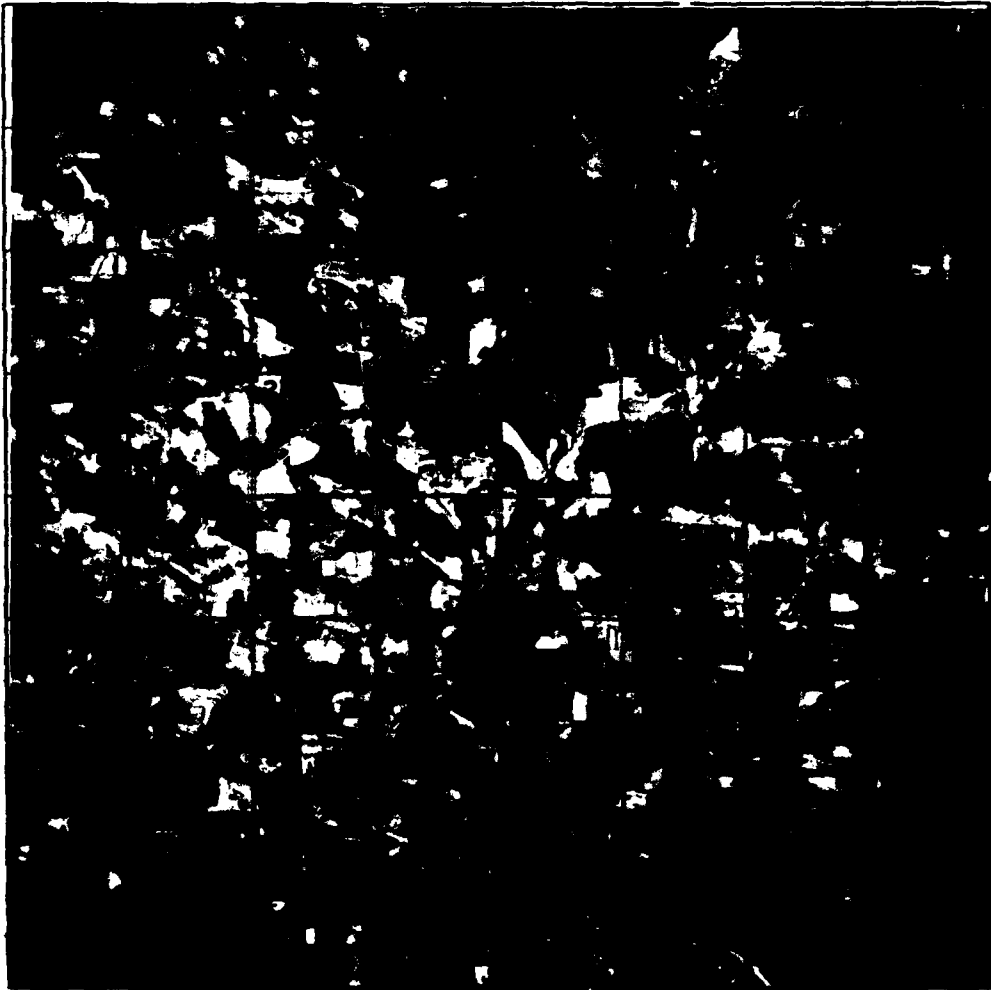


Fig. 5.3: Transparency No. 54

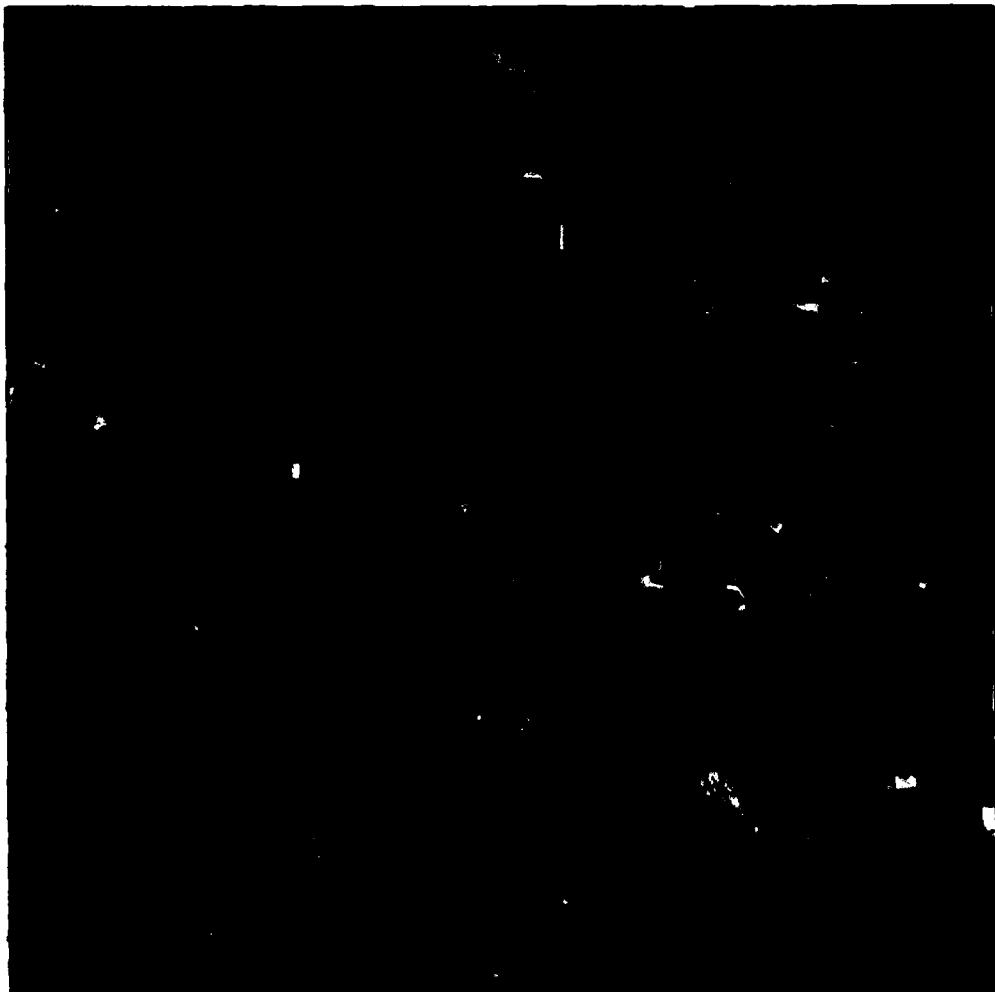


Fig. 5.4: Transparency No. 56

- the identity of area imaged can easily be detected by corresponding parts of the network of roads: segments No. 45, 46, 53 and 54 of fig. 5.5 correspond to segments No. 13, 14, 21, 22, 29 and 30 of fig. 5.6, segment No. 62 of fig. 5.6 corresponds to segment No. 30 of fig. 5.7, etc.
- although the results are comparable globally, there exist considerable local differences: compare segments No. 33, 34, 41 and 42 of fig. 5.5 to segments No. 9, 10, 17 and 18 of fig. 5.6 or segments 38 and 39 of fig. 5.6 to segments No. 6 and 7 of fig. 5.7. The local differences are not only due to the different image quality, but also to the different camera position. This will be confirmed by other results presented in the following
- neither broad lines such as the different lanes of the highway nor very small lines such as the walkways through the forests (upper right part of image No. 52) have been extracted due to the specific values of the parameter "width". Other results will follow which complement these results with regard to line width.

The computer time required to produce these results have been measured and are shown in table 5.1. Individual CPU times of the four main program sections are shown as well as the total CPU time and the total elapsed time which includes CPU time and idle time due to waiting for I/O data transfer. All values of computer time are approximate: although we can tap the internal computer clock, we decided to omit single minutes and seconds, because there is a lot of inferior functions involved in the whole procedure which require a few seconds each and are not worth mentioning. The total elapsed time may, in some cases, be influenced by the time sharing multi job computer operation environment and should be assessed with respect to our computer system only. From the values indicated in table 5.1 it is obvious that:

- a significant amount of CPU time is necessary to process the I/O requirements (i.e. selecting the respective image and result submatrix out of the unsegmented image or result matrix on disk storage)

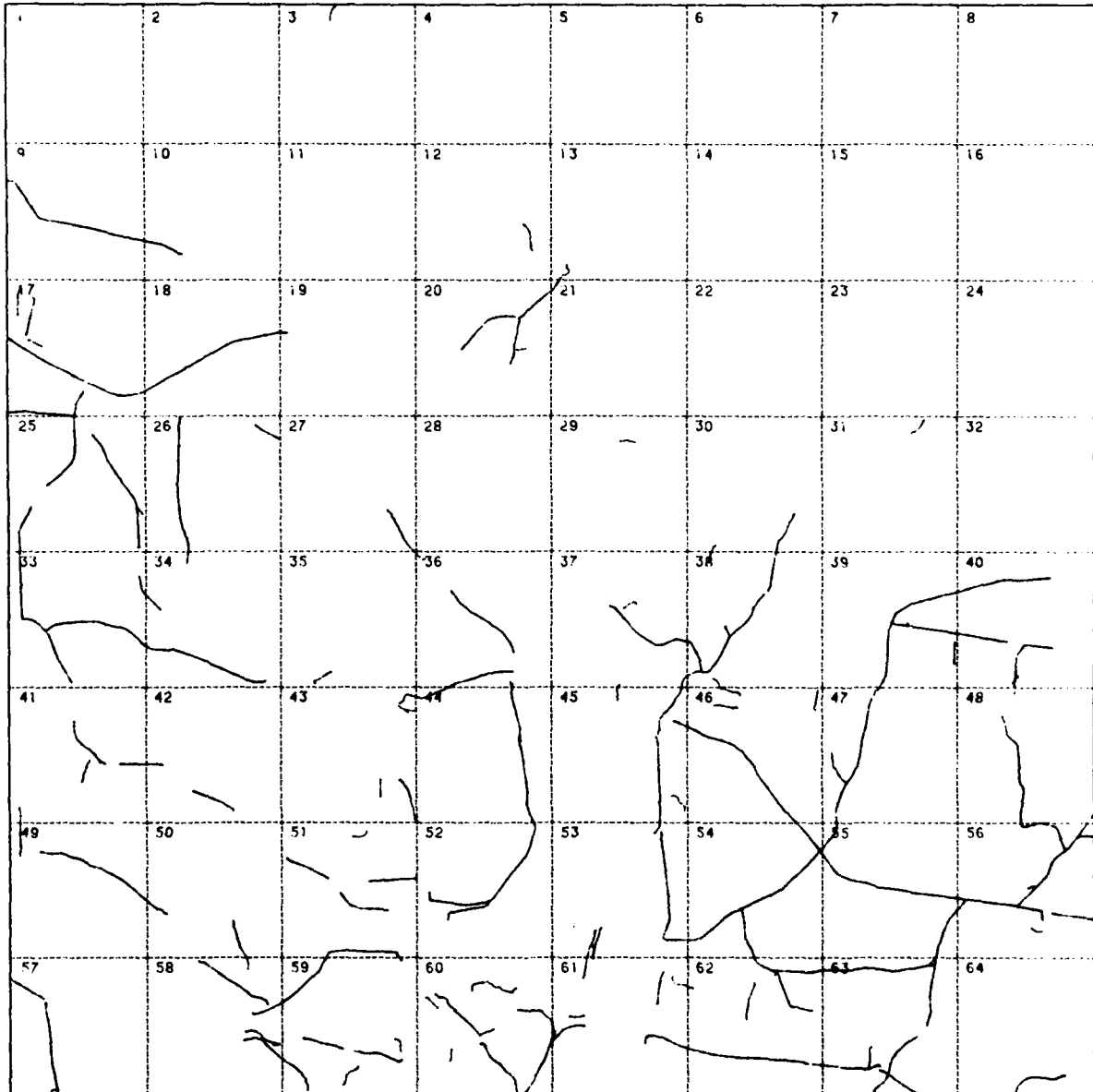


Fig. 5.5: Extraction result from the single image No. 52
using contrast 15 and width 2 to 5

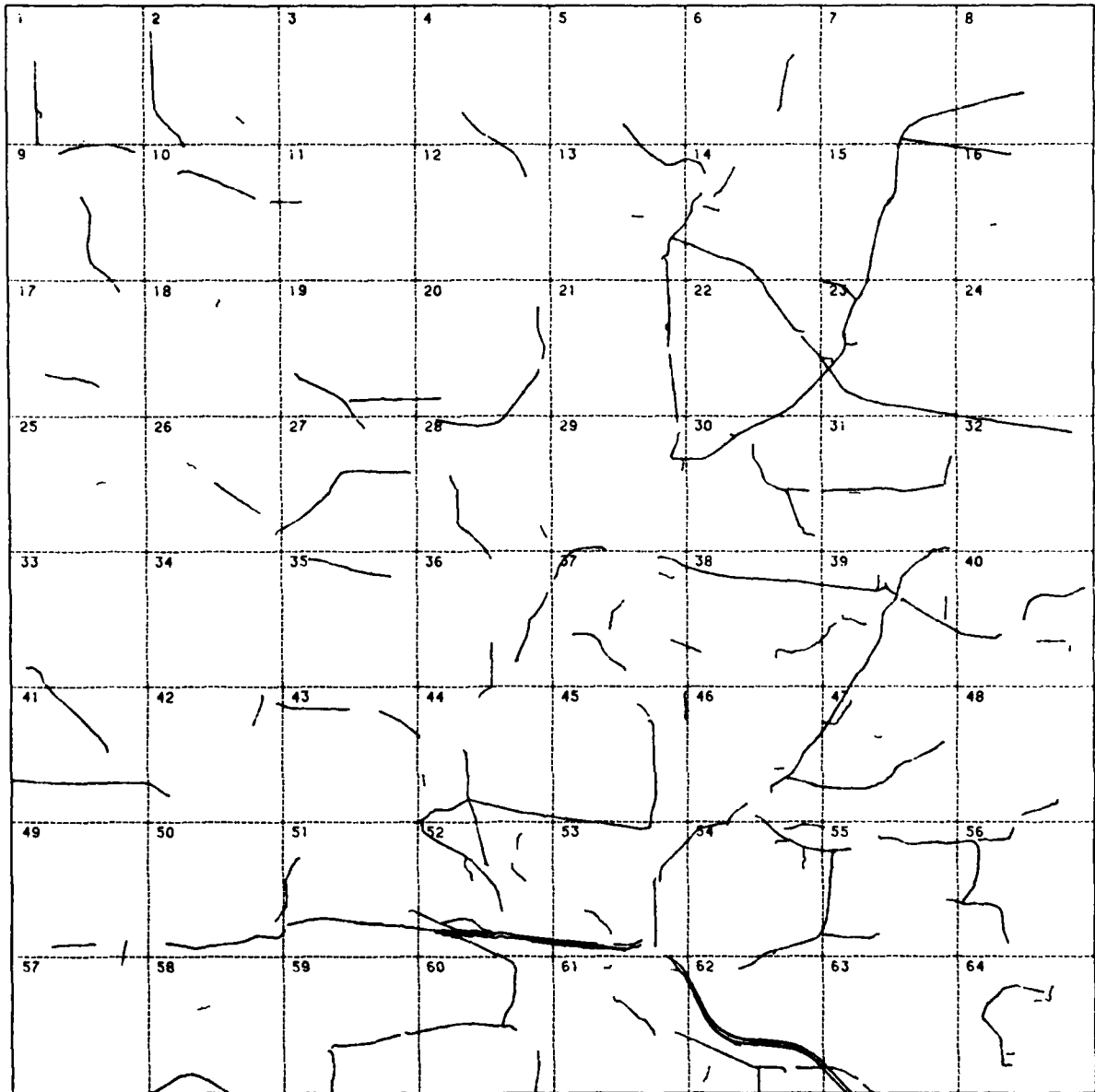


Fig. 5.6: Extraction result from the single image No. 54
using contrast 15 and width 2 to 5

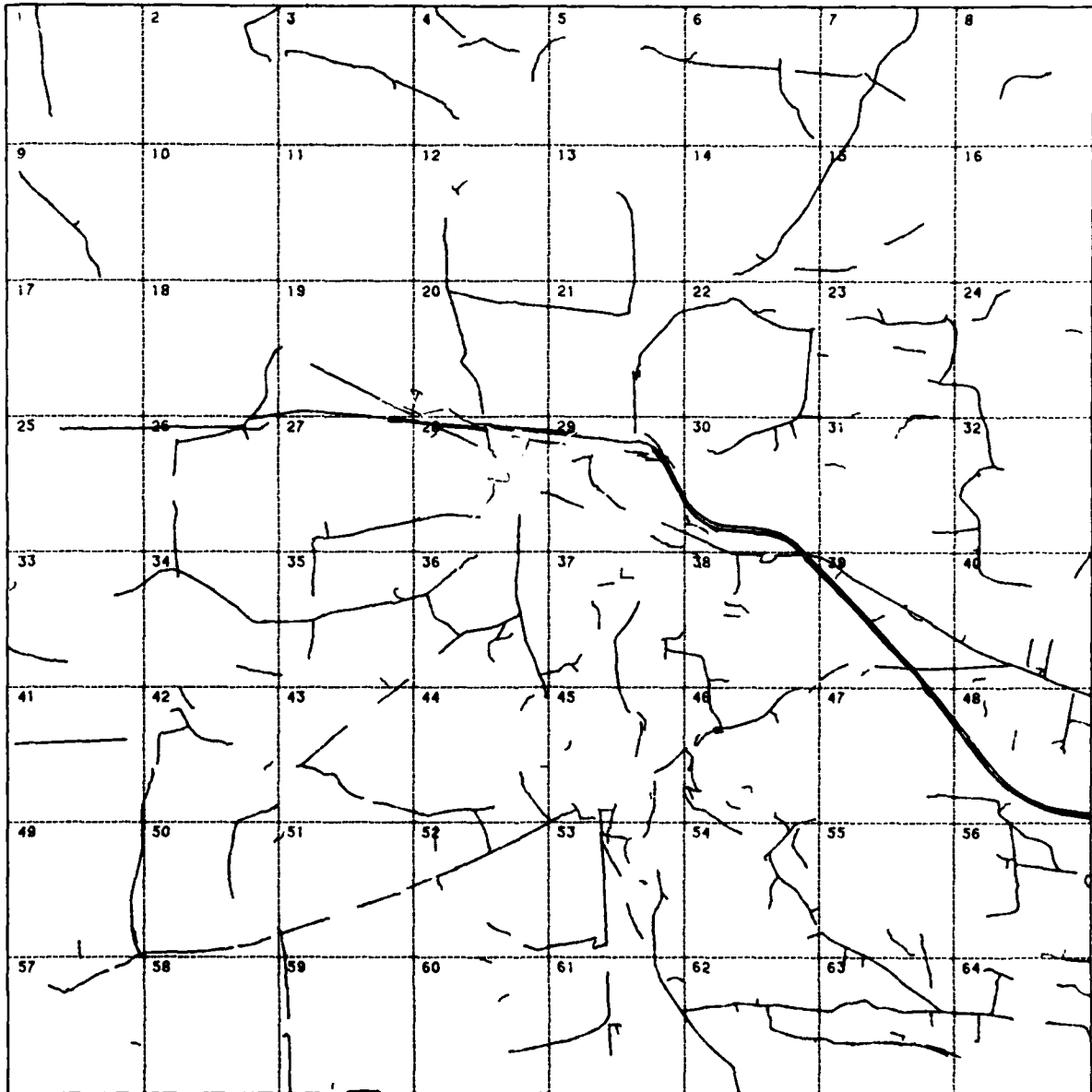


Fig. 5.7: Extraction result from the single image No. 56
using contrast 15 and width 2 to 5

- more than 50% of CPU time required for image data evaluation itself (i.e. - starting point search, IEM and REM) is spent for searching starting points, which is the only function working systematically on the image data. Although there seems to be optimization potential in this function, we feel confirmed in our view to prefer object guided image data evaluation and to avoid systematic, uniform image data evaluation
- although LEM and REM contribute an approximate share of 50% to the total extraction result respectively, LEM requires much less CPU time for its operations, i.e. it operates faster than REM
- the total CPU time amounts to between 4 and 5 hours which is comparable to the values we measured during the previous project /1/, but does not comply with the projected decrease of the CPU time requirement stated at the beginning of this current task. Due to the change of the subimage size from 512 x 512 pixels to 1024 x 1024 pixels we initially achieved a decrease by a factor of two. However, as we discovered later that an unsegmented storage of the image (and result) matrix was necessary and, in consequence, a windowing function to load the respective submatrix was required each time, we lost the initial processing speed advantage
- the total CPU time differs for the three images; this is due to the differing amount of resulting polygons, i.e. for large parts of image No. 52 no line object with the properties specified were detected; hence, the whole process was faster
- the total elapsed time extends to approx. twice the amount of the total CPU time. This indicates that a lot of idle time is wasted by waiting for the data to be loaded from or stored onto disk. A virtual addressing technique, offered by DEC and other computer manufacturers, could cut down these idle times considerably.

The following sequence of experiments refers to image No. 56 as input data, because this image contains the most dense network of lines. The first experiment addressed the different possibilities to complete extraction results from an image by additional evaluation of the same image with different parameter values. Figure 5.8 shows the result of a continuation extraction from image No. 56; the contrast requirement was set to 15 gray levels as before, the width tolerance was changed to 4 to 6 pixels; the extraction was performed on the basis of

the result shown in fig. 5.7, i.e. only additional results not already contained in the result matrix were generated by starting the extraction procedure at every dead end of that run and at every new starting point found in addition to the existing result. It can be seen from fig. 5.8, that a lot of short continuations and additions was found under these extraction conditions. The computer times for this run are shown in line four of table 5.1: image data transfer processing time of 85 min again took 40% of the total CPU time, which amounted to 205 min, starting point processing time of 50 min amounted to one half of the previous value only, obviously because not many additional starting points could be found, and the total elapsed time again amounted to approx. twice the quantity of the total CPU time. Fig. 5.9 shows an overlay of the result within the line width tolerance of 2 to 6 pixels. Comparing figures 5.7 and 5.9, one can notice the completion and the connection of parts previously separated (see segments No. 4, 6, 9, 19, 36, etc.). However, within the boundaries of the city in the middle of the image, quite a lot of short and disconnected resulting line segments still exists. This problem was no subject of the current research project; as far as we know, no satisfying solution to this problem seems to exist.

To complement the experiment described above, we performed a separate extraction from image No. 56 with the same parameter values, i.e. a contrast requirement of 15 gray levels and a width tolerance of 4 to 6 pixels. The result of this run did not update a result matrix previously filled; the run started from a blank result matrix and searched its own starting points (see fig. 5.10). The resulting lines consist of several extended parts of the line network; many parts of this result are identical to the corresponding parts of the result shown in figure 5.7, because the width tolerances were overlapping in both cases. It is also obvious that the significant parts of the continuation result shown in fig. 5.8 are covered by the result of this separate extraction, because the width tolerances were identical in both cases.

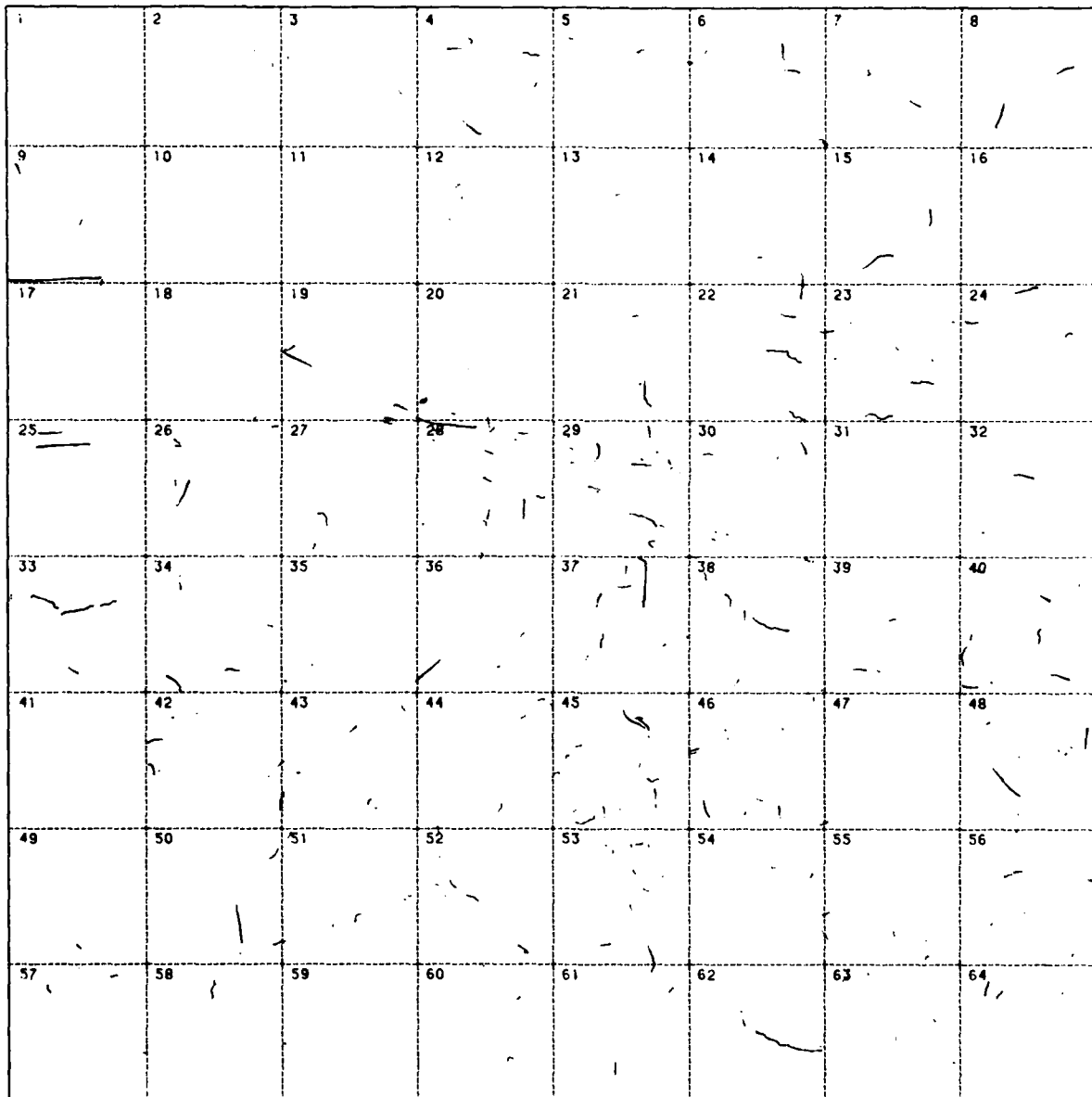


Fig. 5.8: Extraction result from the single image No. 56 using contrast 15 and width 4 to 6, as a continuation to the result of fig. 5.7

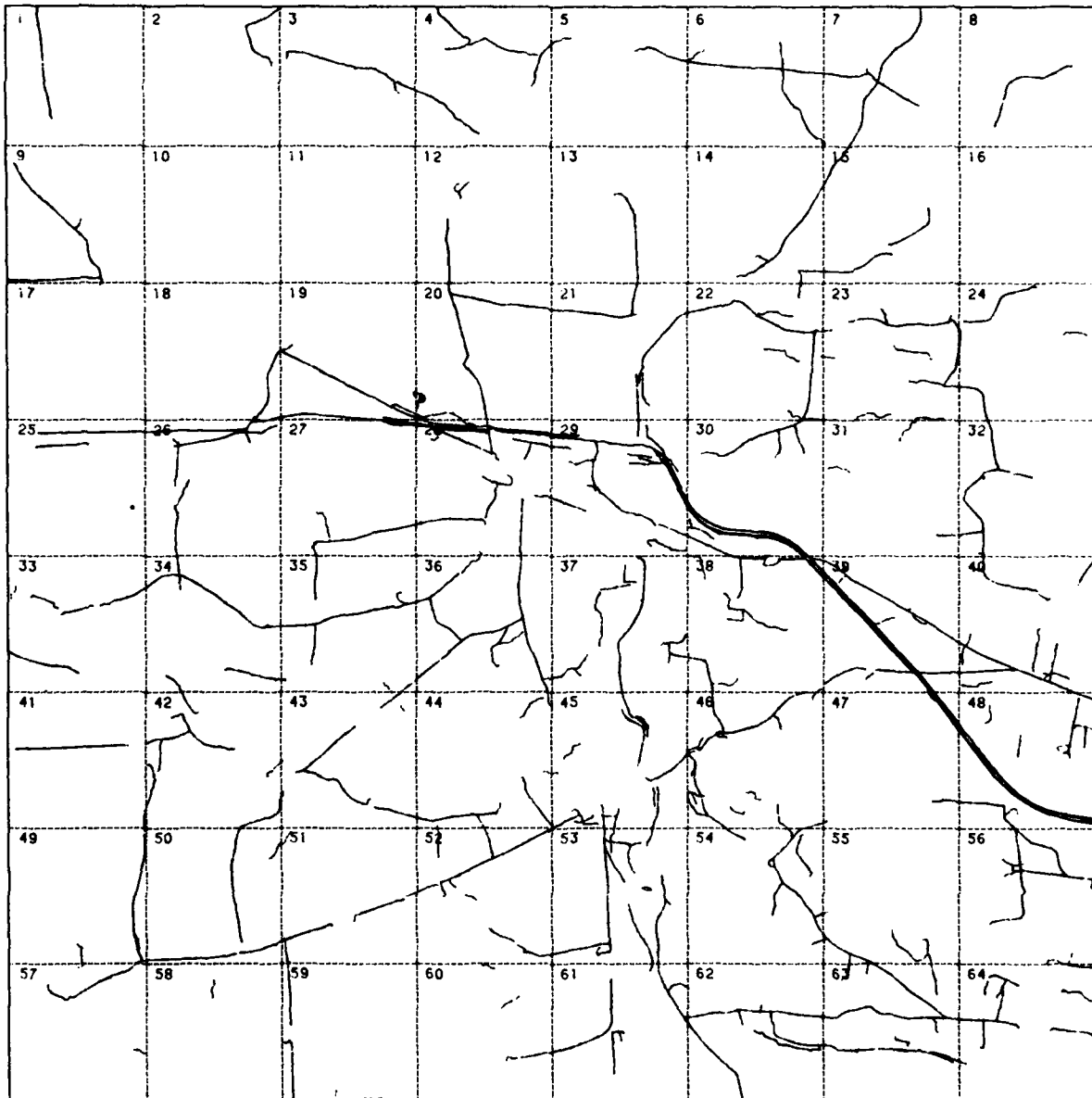


Fig. 5.9: Composite of the results of fig. 5.7 and 5.8

The processing time for the result shown in fig. 5.10 is indicated in line five of table 5.1. The total CPU time of 185 min is less than the amount required for the continuation extraction. This decrease is obviously due to the high degree connectivity of the extraction result. Remember that a break off dead end always is produced by REM; the big number of dead ends shown in fig. 5.8 (continuation extraction) requires a big share of REM activity (60 min CPU time), whereas a comparatively small number of dead ends of the connected line network shown in fig. 5.10 (separate, initial extraction with identical line parameters) requires less REM activity (30 min CPU time). The amount of image data transfer processing time is again in the same range (95 min CPU time).

Figure 5.11 shows the overlay of the results from fig. 5.7 and fig. 5.10. Theoretically, this composite should be identical to the composite shown in fig. 5.9, although they were generated in different ways. Indeed, with the exception of a few locations (segment No. 8, 9, 15, 18 and 61) the results are identical!

The presentation of extraction results from single images is concluded by a sequence addressing the impact of the contrast threshold. Figures 5.12, 5.13 and 5.14 show extraction results from image No. 56, using a constant line width tolerance of 2 to 6 pixels, but an increasing contrast requirement of 15, 20 and 25 gray levels respectively. Comparing these results the following remarks can be stated:

- the basic network of roads is identical in all three results; this proves that the extraction procedure is not too sensitive to differing contrasts
- with increasing contrast requirements a slightly decreasing amount of extraction results is generated; especially the number of short, isolated extraction results decreases; the degree of connectivity of the extended parts of the results remains constant, approximately

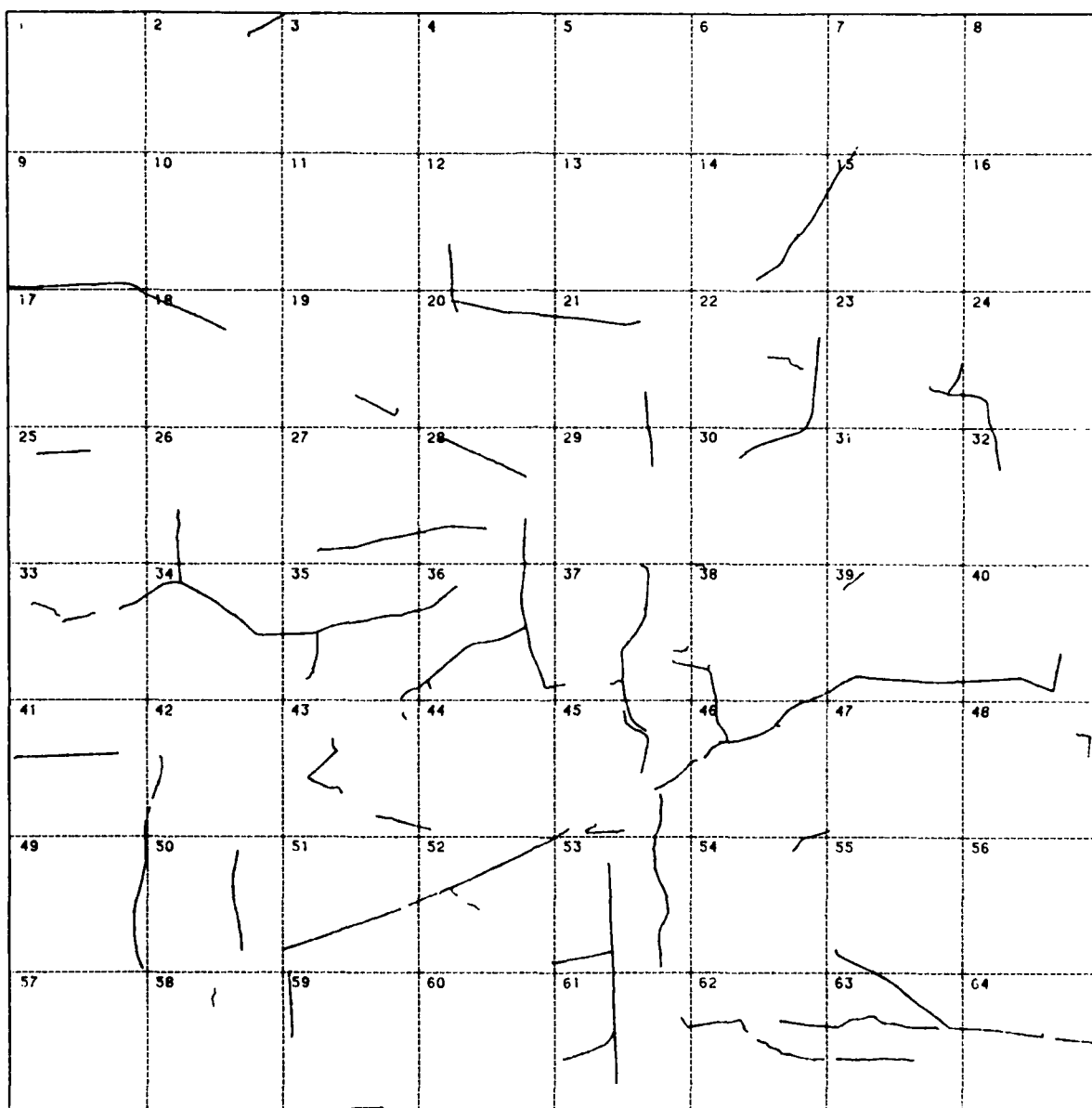


Fig. 5.10: Extraction result from the single image No. 56 using contrast 15 and width 4 to 6, as a separate run

- with increasing contrast requirements the processing time (indicated in lines 6 through 8 of table 5.1) decreases significantly from 335 min via 285 min to 250 min of total CPU time. With an approximately constant amount of time for image data transfer processing, the processing time required for the starting point search and for REM decreases significantly. This reflects the fact that - with increasing contrast requirements - a decreasing number of starting points satisfies the requirements (thus eliminating some false starting points, because the majority of lines is still extracted) and a decreasing number of short, isolated line segments with dead ends at each end necessitates REM activity.

Finally, the results shown in figures 5.9, 5.11 and 5.12 deserve comments, because all three should display identical results, generated in three different ways:

- identical image data
- identical parameter values (contrast requirement of 15 gray levels, line width tolerance of 2 to 6 pixels)
- identical extraction procedure; however, the first two results are composites of two extraction results each, whereas the third result was produced by a single extraction run.

In fact, differences between those three extraction results are hardly noticeable: see segments No. 8, 9 and 35 for differences in contrast to the majority of the segments which contain results of a very high degree of similarity. The generation of the third result required the smallest amount of processing time (335 min of total CPU time), because the other results required two extraction runs each. This proves that, within reasonable limits of line width tolerance, the extraction procedure is nearly insensitive to varying line width, which renders the generation of composite results by different, selective extraction runs dispensable.

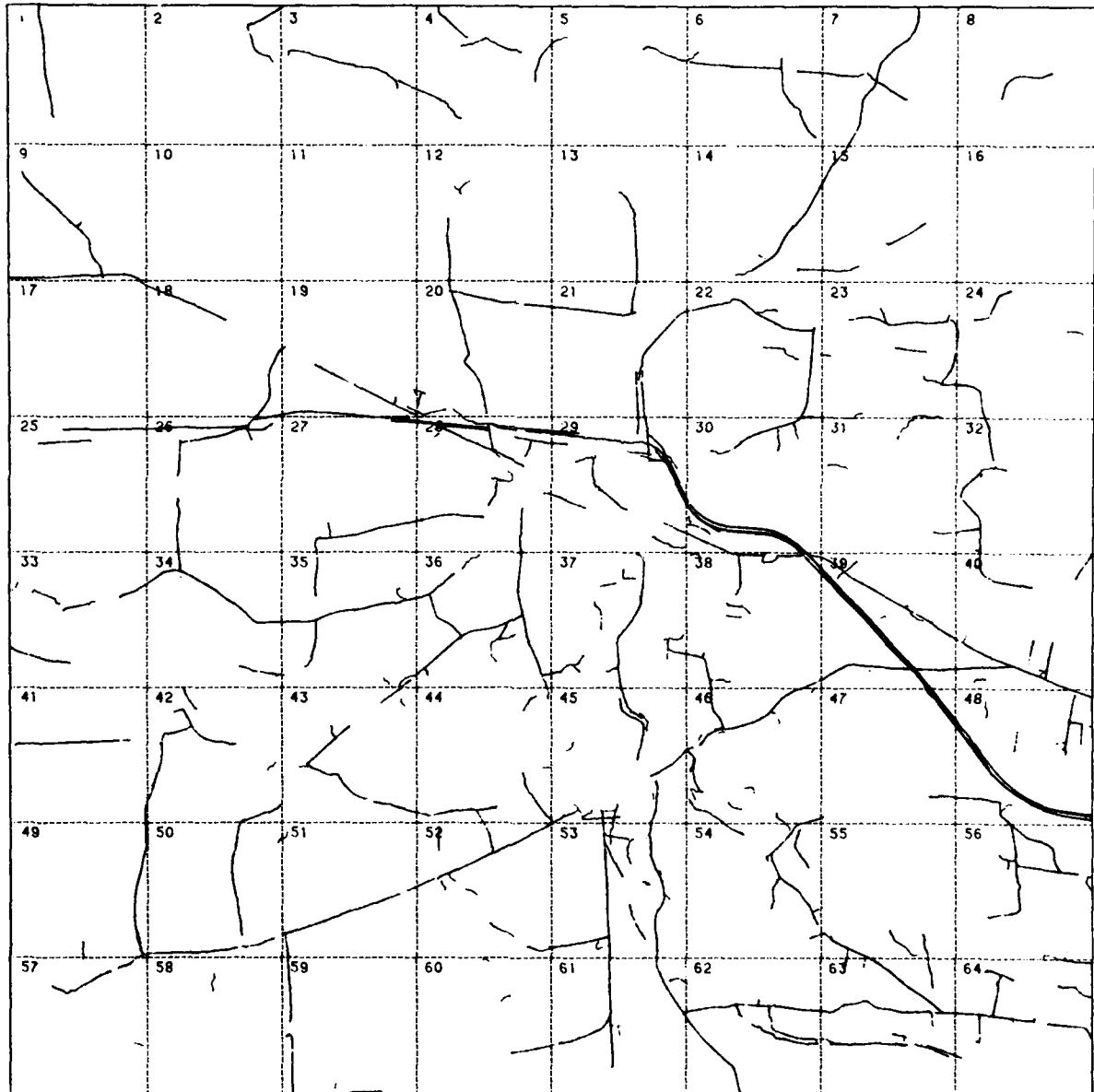


Fig. 5.11: Composite of the results of fig. 5.7 and 5.10

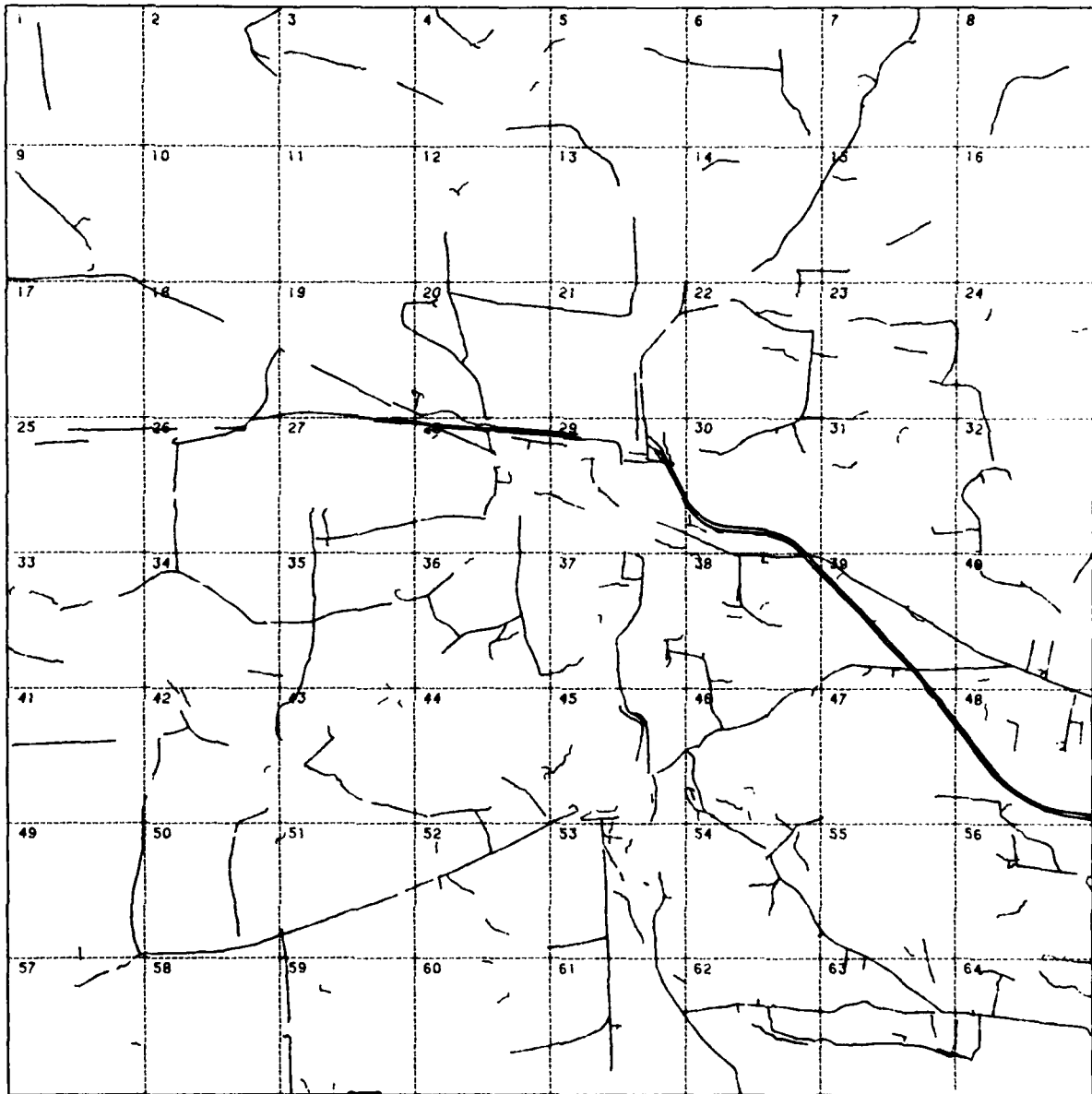


Fig. 5.12: Extraction result from the single image No. 56
using contrast 15 and width 2 to 6

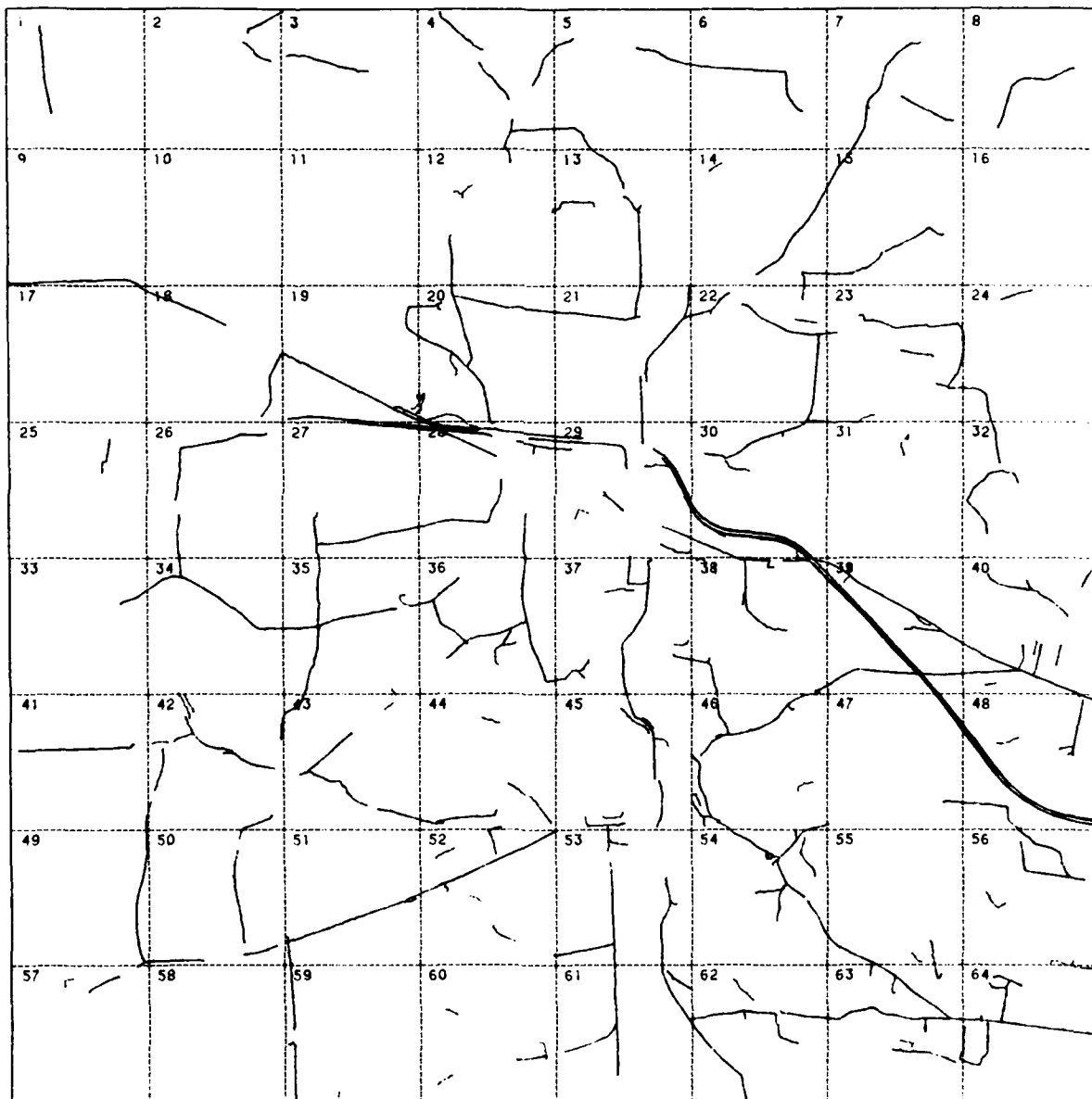


Fig. 5.13: Extraction result from the single image No. 56
using contrast 20 and width 2 to 6

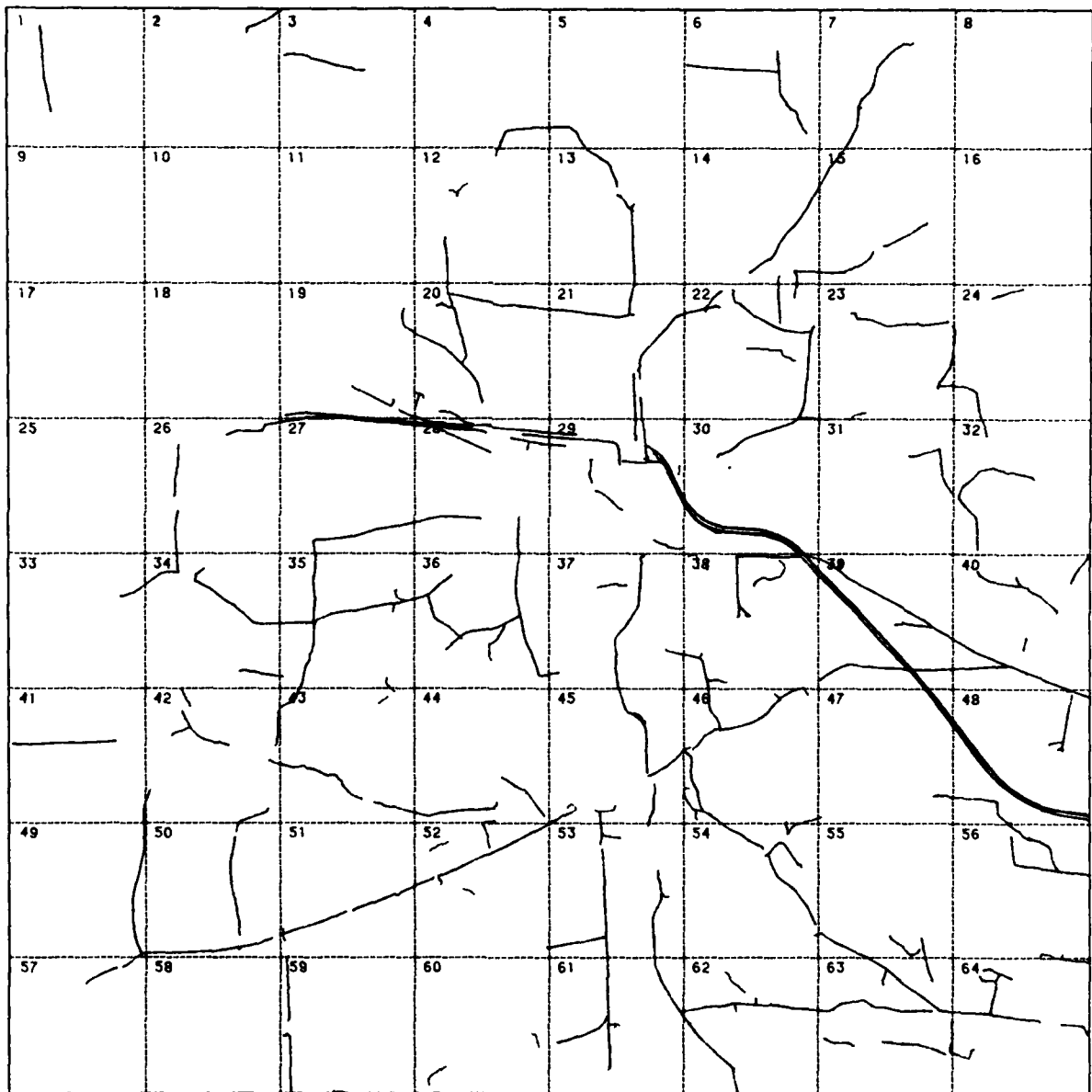


Fig. 5.14: Extraction result from the single image No. 56
using contrast 25 and width 2 to 6

Approximate times in minutes	Image data transfer proc.	Starting point search	LEM	REM	Total CPU time	Elapsed time
Image No. 52 contrast 15 line width 2 to 5	85	100	10	45	245	500
Image No. 54 contrast 15 line width 2 to 5	95	90	10	60	260	560
Image No. 56 contrast 15 line width 2 to 5	105	95	20	85	310	625
Image No. 56 contrast 15 line width 4 to 6 as continuation	85	50	5	60	205	450
Image No. 56 contrast 15 line width 4 to 6 initial extraction	95	50	5	30	185	470
Image No. 56 contrast 15 line width 2 to 6	105	105	20	100	335	660
Image No. 56 contrast 20 line width 2 to 6	105	75	20	80	285	650
Image No. 56 contrast 25 line width 2 to 6	105	60	15	65	250	570

Table 5.1: Processing times for the extraction experiments in single images

5.3 Results of coincident extraction in stereo image pairs of 4000 x 4000 pixels

Two different arguments influenced the decision to apply the procedure of coincident line extraction from images of a stereo pair to an image size of 4000 x 4000 pixels:

- by extrapolating the processing time measured for the evaluation of single images of 8000 x 8000 pixels, we calculated a total amount of processing time for the procedure of coincident extraction, which would not allow its application to images of 8000 x 8000 pixels
- the area covered by both images of each stereo pair is limited to 4200 x 8000 pixels approximately; it would, therefore, be a waste of processing time to evaluate a complete frame of 8000 x 8000 pixels, where only half of the result could have been mapped into the stereo partner image for verification and update.

Thus, we decided to limit the stereo evaluation to the central part of the overlapping area of a stereo pair: we selected corresponding sections of the images No. 54 (see fig. 5.15) and No. 56 (see fig. 5.16); the size of each section was chosen to amount to 4000 x 4000 pixels exactly, which equals 4 x 4 segments of 1024 x 1024 pixels each, with 32 pixels of overlapping between the segments. The stereo pair of images No. 54 and 56 was chosen because of the density of the line network in the overlapping area. To show the various effects of differing image quality, we decided to run two experiments of coincident extraction from one stereo pair of images by starting the procedure in image No. 54 and starting the procedure in image No. 56. The results will be shown subsequently. Processing times have been measured and will be explained.

The first run of coincident extraction starts the initial extraction from image No. 54. The extraction parameters were set to standard values:

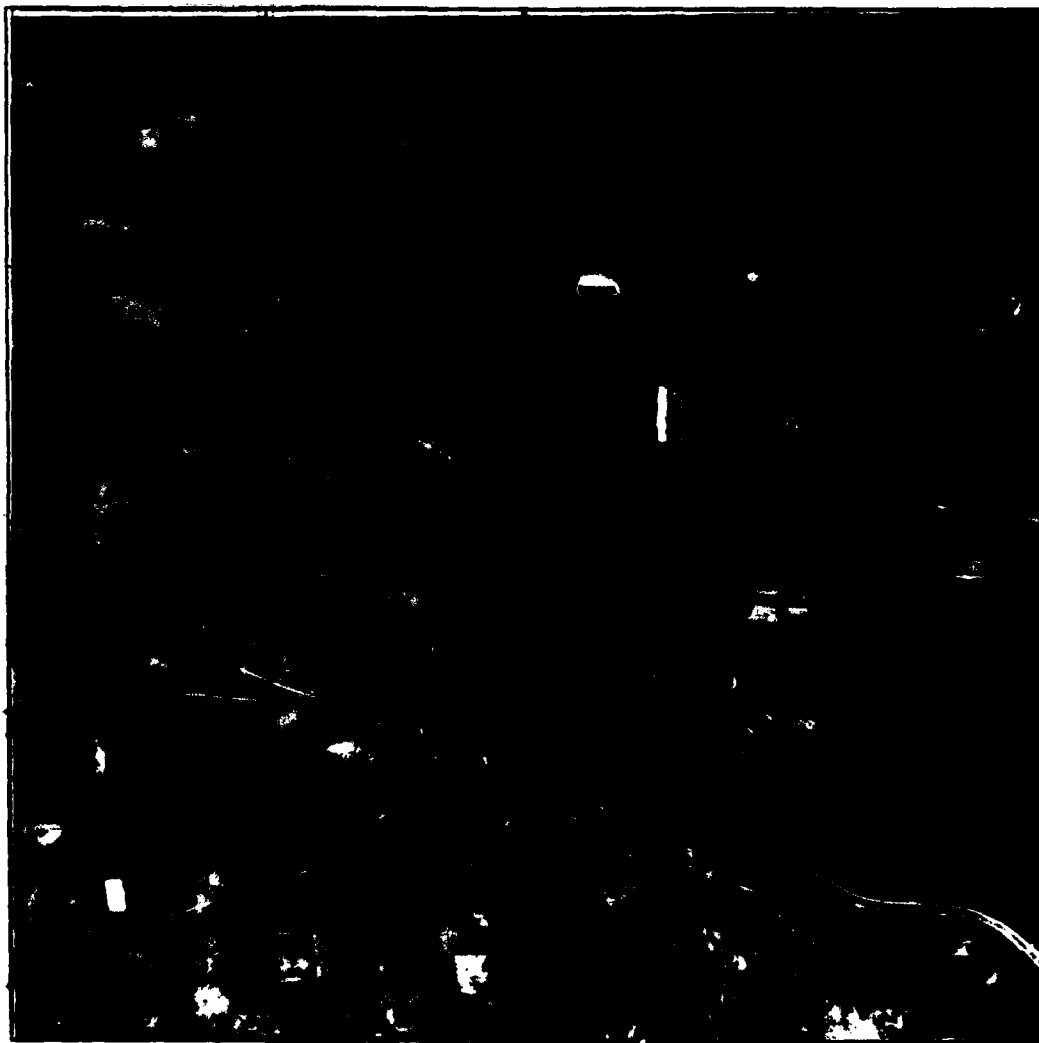


Fig. 5.15: Central section of image No. 54 to be used as
"left image" of the stereo pair

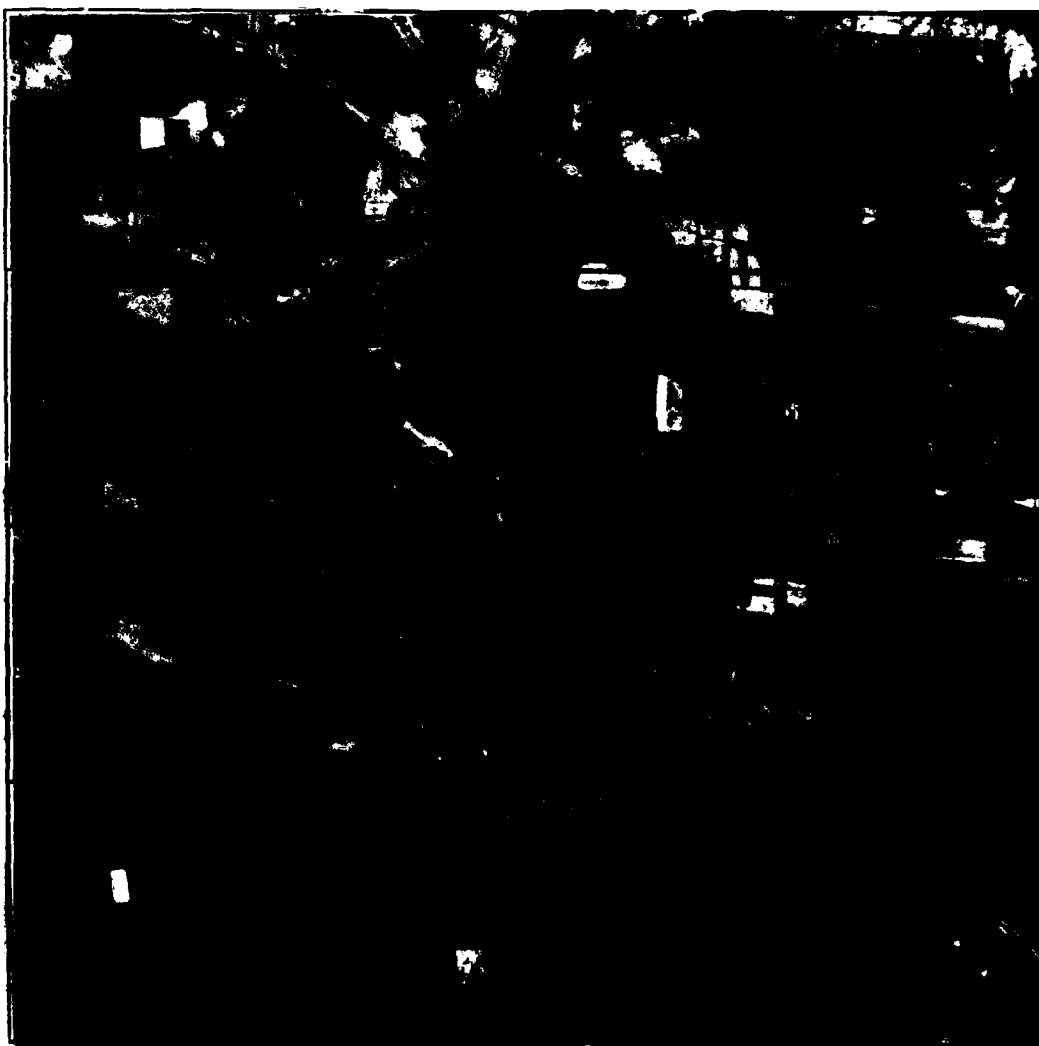


Fig. 5.16: Central section of image No. 56 to be used as
"right image" of the stereo pair

- contrast type "peak" (bright object)
- contrast requirement 15 gray levels
- width tolerance 2 to 5 pixels

Fig. 5.17 shows the result of this process. Due to the statistical aspects of the starting point search, this result is not totally identical to the corresponding part of the result shown in fig. 5.6. However, the basic parts of the line network are identical. According to the philosophy of the coincident extraction procedure, we consider these results as preliminary and try to verify them by evaluation of the stereo partner image. The results of the processing steps required for the purpose of verification, do not reveal much information, but are shown here for reasons of completeness:

- the output of the maximum length polygon compilation (see fig. 5.18) has an identical display, but a different internal format as its input data
- the output of the mapping process also has the same display (see fig. 5.19), but it is now related to the coordinate system of the stereo partner image No. 56.

The verification process evaluates image No. 56 along the predicted lines as shown in fig. 5.19. The result of this process is shown in fig. 5.20: these are the lines predicted from image No. 54 to exist in image No. 56 and accepted for image No. 56 after evaluation of that image data. Notice the difference between the input (fig. 5.19) and the output (fig. 5.20) of the verification process: some small parts of the predicted lines were not accepted, which caused some additional gaps and disconnections in the network of lines. At that time the corresponding network of accepted lines for image No. 54 has been generated in the computer; it will be shown later, when we are back to display results related to the coordinate system of this image. Before that, we want to complete the network of lines by continuing the extraction process at each dead end

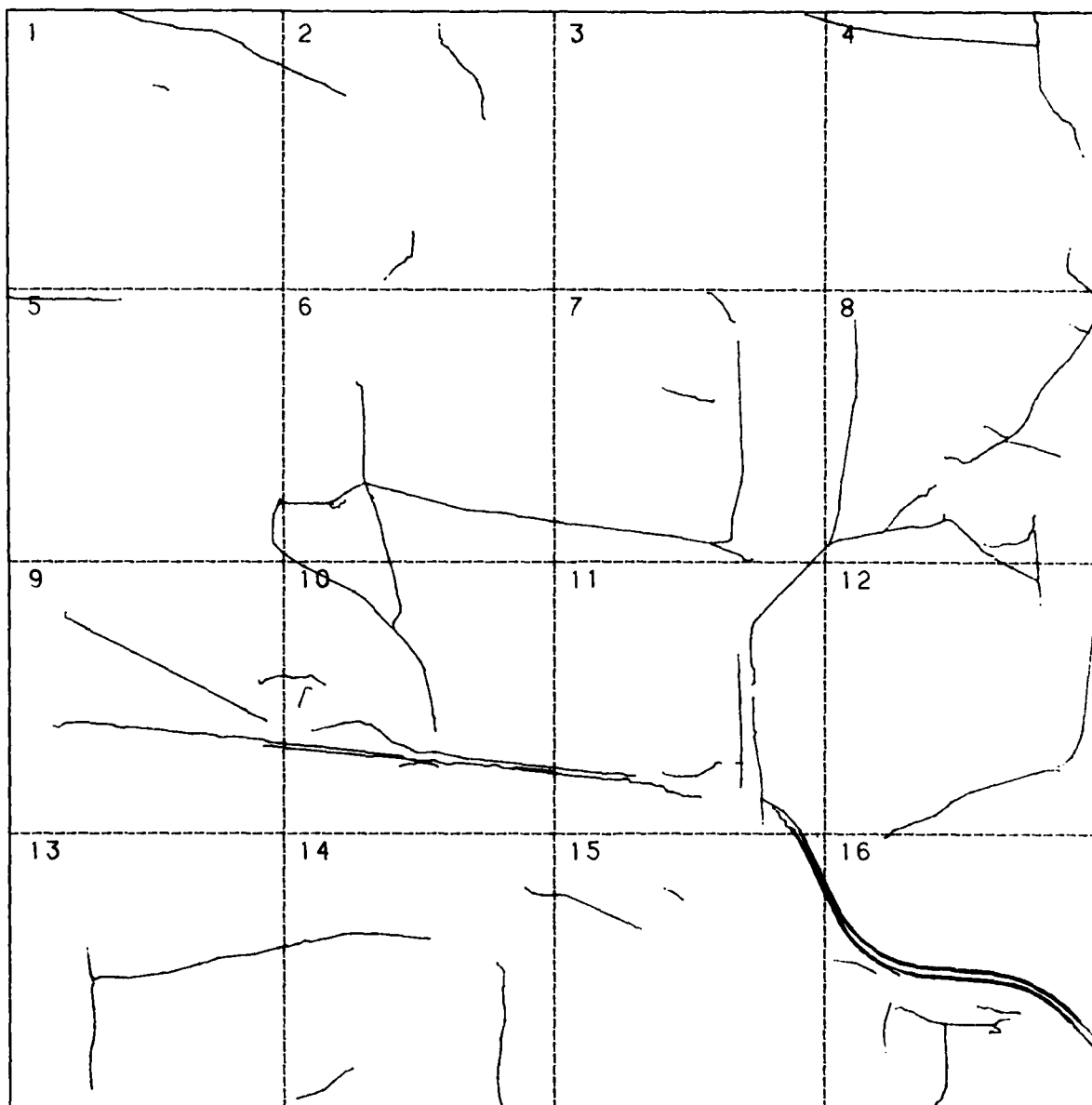


Fig. 5.17: Result of initial extraction from image No. 54
as shown in fig. 5.15



Fig. 5.18: Result of maximum length polygon compilation
of fig. 5.17

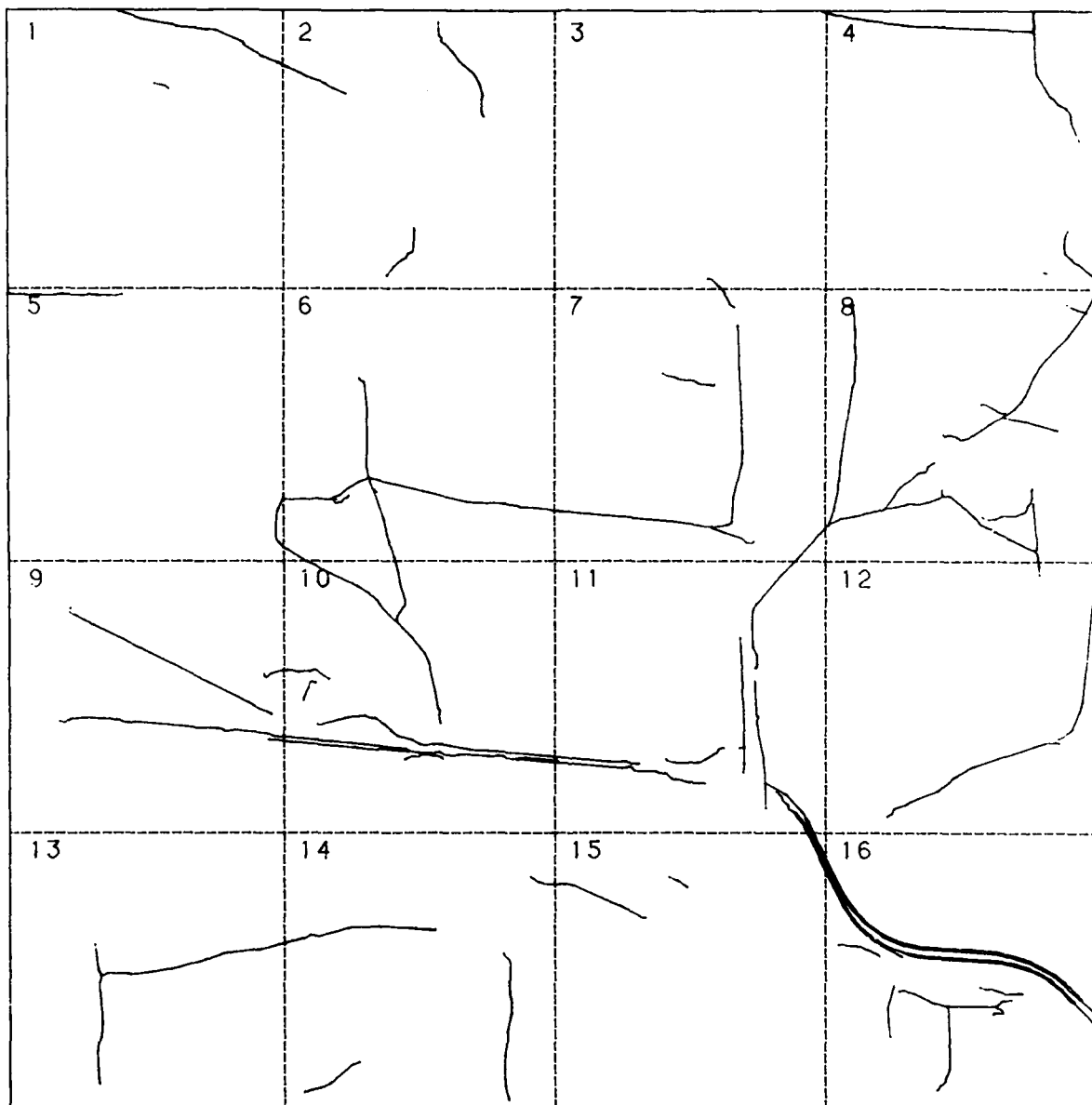


Fig. 5.19: Result of mapping the maximum length polygons
of fig. 5.18 to the coordinate system of image
No. 56 as shown in fig. 5.16

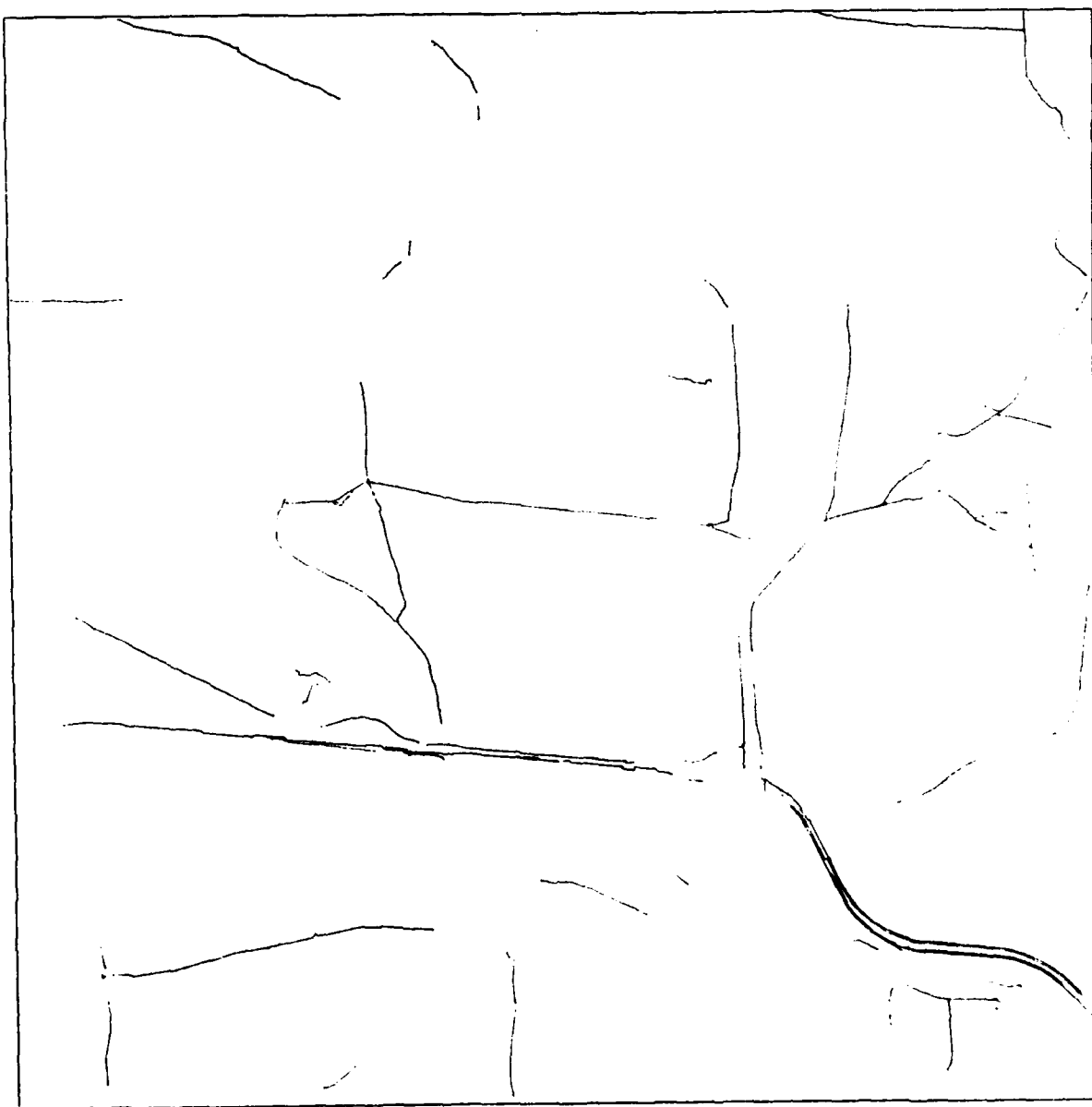


Fig. 5.20: Result of the first verification process
performed in image No. 56



Fig. 5.21: Result of the first continuation extraction
performed in image No. 56

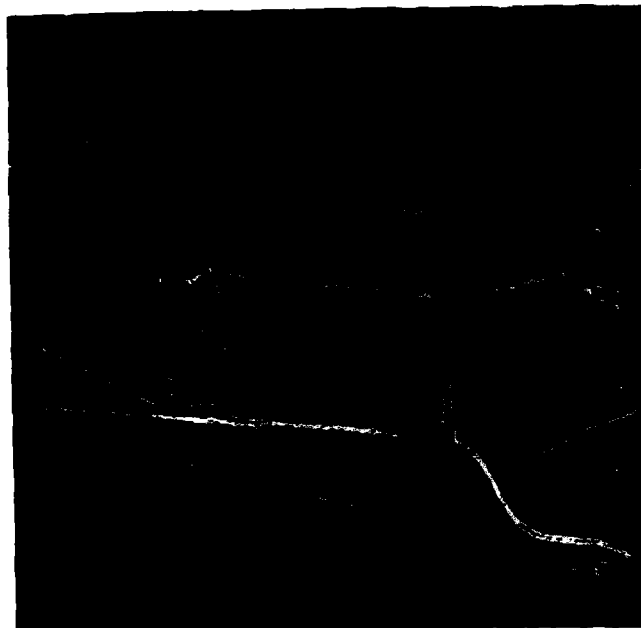


Fig. 5.22: Composite of the result of the first verification (yellow lines) and the result of the first continuation extraction (blue lines)

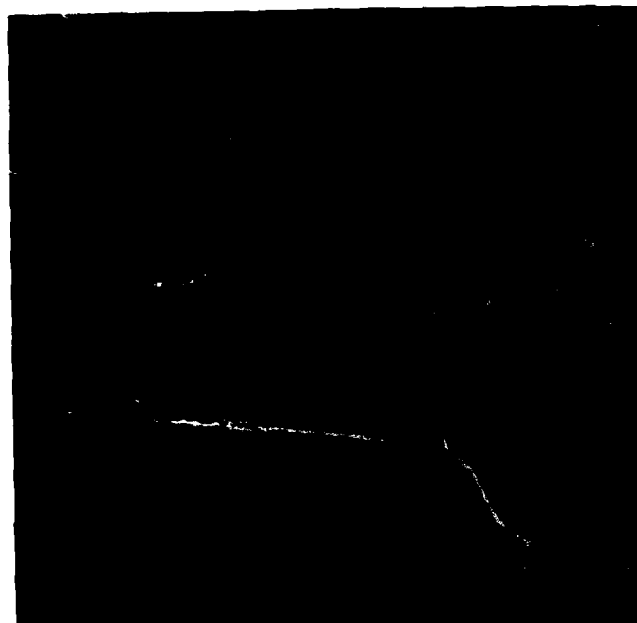


Fig. 5.24: Composite of the result of the first verification (yellow lines) and the result of the second verification (violet lines), both related to image No. 54

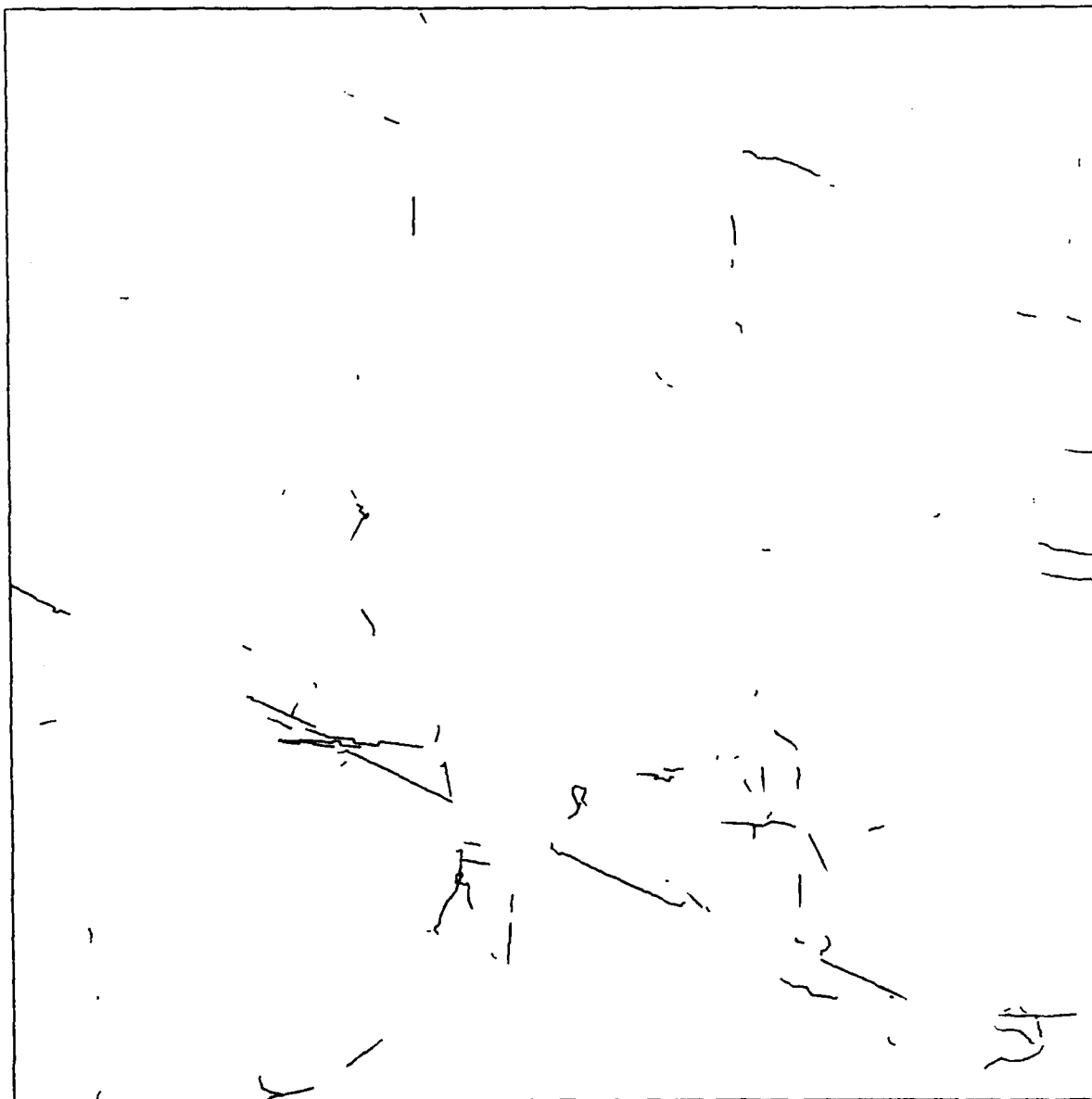


Fig. 5.23: Result of the second verification process performed
in image No. 54

and at new starting points in image No. 56. The extraction parameters are the same as mentioned above. The result of this continuation extraction shown in fig. 5.21 promises to be a significant contribution to the resulting network of lines. (Up to this moment, it is a promise only, because this continuation result has to pass the verification process, yet). To demonstrate the possible completion and increasing connectivity of the extraction result we generated a color print by overlaying the first verification result (fig. 5.20) in yellow and the (preliminary) result of the first continuation extraction (fig. 5.21) in blue as shown in fig. 5.22. The output of the graphic result on a raster screen, together with the photo process, does not produce the same print quality as a color plot; however, it illustrates very well the complementary steps of the extraction procedure.

The next steps in preparation of the second verification process are maximum length polygon compilation and mapping into the coordinate system of image No. 54; their results are not shown here. The result of the verification itself is shown in fig. 5.23. By comparing fig. 5.21 (the first continuation of the extraction) and fig. 5.23 (the accepted parts of this continuation extraction result) one can notice that the majority of lines has been accepted and only short segments have been rejected. Remember that fig. 5.23 shows true results for image No. 54, predicted by extraction from image No. 56, generated by evaluation of image No. 54 and related to its coordinate system. Accordingly, the corresponding parts of accepted results are generated for image No. 56, and will be shown later.

As a second color overlay, we present the composite of this verification result in violet color with the result of the first verification process (in yellow), both related to the coordinate system of image No. 54. It can be seen that there is not much difference to the result shown in fig. 5.22, the

blue part of which lacked verification. It remains to be stressed that the first extraction result from an aerial image (image No. 54 in our example), independent of the power of the extraction methods, can be completed by another evaluation of the same image data via the proper prediction of high probability locations.

And the procedure is not at its end, yet. A second continuation of the extraction, performed on the data of image No. 54 follows. The result shown in fig. 5.25 may still be a remarkable contribution to the final extraction result. Skipping the presentation of the results of maximum length polygon compilation and mapping into the corresponding coordinate system, we present the verification result as the actual contribution of this step to the final result (see fig. 5.26). At this stage of the whole extraction procedure we decided to stop the evaluations. A third continuation extraction was initiated to work on dead ends still existing in image No. 56; the result of this run shown in fig. 5.27 was not submitted to verification and, hence, not incorporated in the final result.

Thus, at the end of these three iterations of extraction and verification processes, the final results for both images of the stereo pair will be available as shown in fig. 5.28 for image No. 54 and fig. 5.29 for image No. 56. These results need the following comments:

- fig. 5.28 shows, in yellow, extraction results from image No. 54 which have been confirmed by verification in image No. 56, and, it shows, in blue, verification results of image No. 54, which have been extracted from image No. 56 to be used as a prediction of their existence in image No. 54. In other words, all these results are generated by direct evaluation of the data of image No. 54 and they are influenced (predicted or verified) by the evaluation of the data of image No. 56. All these results are related to the coordinate system of image No. 54



Fig. 5.25: Result of the second continuation extraction
performed in image No. 54

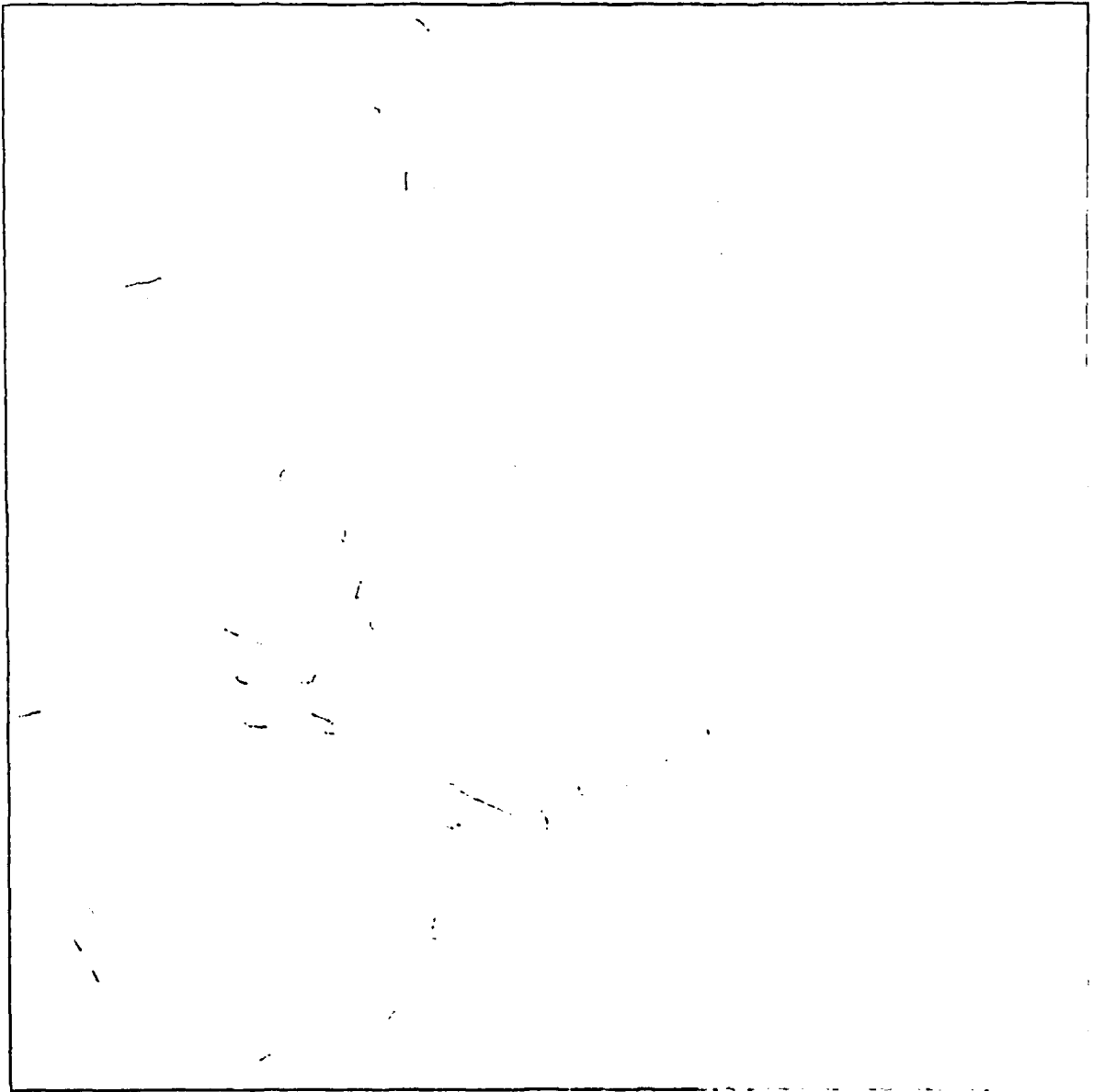


Fig. 5.26: Result of the third verification process performed
in image No. 56



Fig. 5.27: Result of the third continuation extraction performed in image No. 56, which is not added to the final result

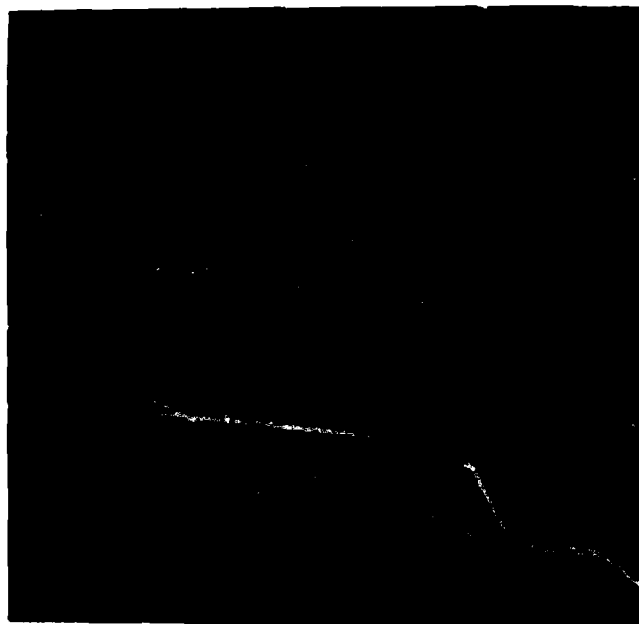


Fig. 5.28: Final result for image No. 54 consisting of extraction results from image No. 54 (yellow) and prediction results from image No. 56 (blue)



Fig. 5.29: Final result for image No. 56 consisting of prediction results from image No. 54 (yellow) and extraction results from image No. 56 (blue)

- correspondingly, fig. 5.29 shows, in yellow, verification results from image No. 56, which have been extracted from image No. 54 to be used as a prediction of their existence in image No. 56, and it shows, in blue, extraction results from image No. 56 which have been confirmed by verification in image No. 54. In other words, all these results are generated by direct evaluation of the data of image No. 56 and they are influenced (predicted or verified) by the evaluation of the data of image No. 54. All these results are related to the coordinate system of image No. 56
- besides this correspondence of the results, one can notice a considerable completion and increase of connectivity of the results by comparing the final results (e.g. fig. 5.28) to the initial extraction (fig. 5.17).

The processing times required for this procedure of coincident line extraction are indicated in table 5.2. From this table it can be seen that:

- for images of 4000 x 4000 pixels, CPU time requirements came down to reasonable amounts
- in general, CPU times decrease for the higher numbered iteration steps, due to the decreasing amount of results
- the total elapsed time is significantly extended by data input and output.

For comparison, a second experiment of coincident line extraction from the same stereo image pair has been performed, starting from image No. 56. The same sequence of processing steps has been initiated which generated the following sequence of results:

- fig. 5.30 shows the result of the initial extraction from image No. 56. Note the difference to fig. 5.17, espec. segments No. 1, 3, 5, 6, 8, 9, 14, 16. Besides the different camera position, differing image quality must be responsible for these differing results
- fig. 5.31 shows the result of the maximum length polygon compilation
- fig. 5.32 shows the result of mapping the polygons into the coordinate system of image No. 54

Process	CPU time	Elapsed time
Initial extraction	60	120
First MLP compilation	35	150
First mapping	0.1	0.2
First verification	95	325
First continuation extraction	65	115
Second MLP compilation	20	100
Second mapping	0.1	0.3
Second verification	95	520*
Second continuation extraction	25	65
Third MLP compilation	15	70
Third mapping	0.1	1
Third verification	55	285*
Third continuation extraction	15	85
Final result processing	125	585*

Table 5.2: Processing times for coincident extraction starting from image No. 54 (approximate time in minutes)

* process was running during day time under computer time sharing conditions

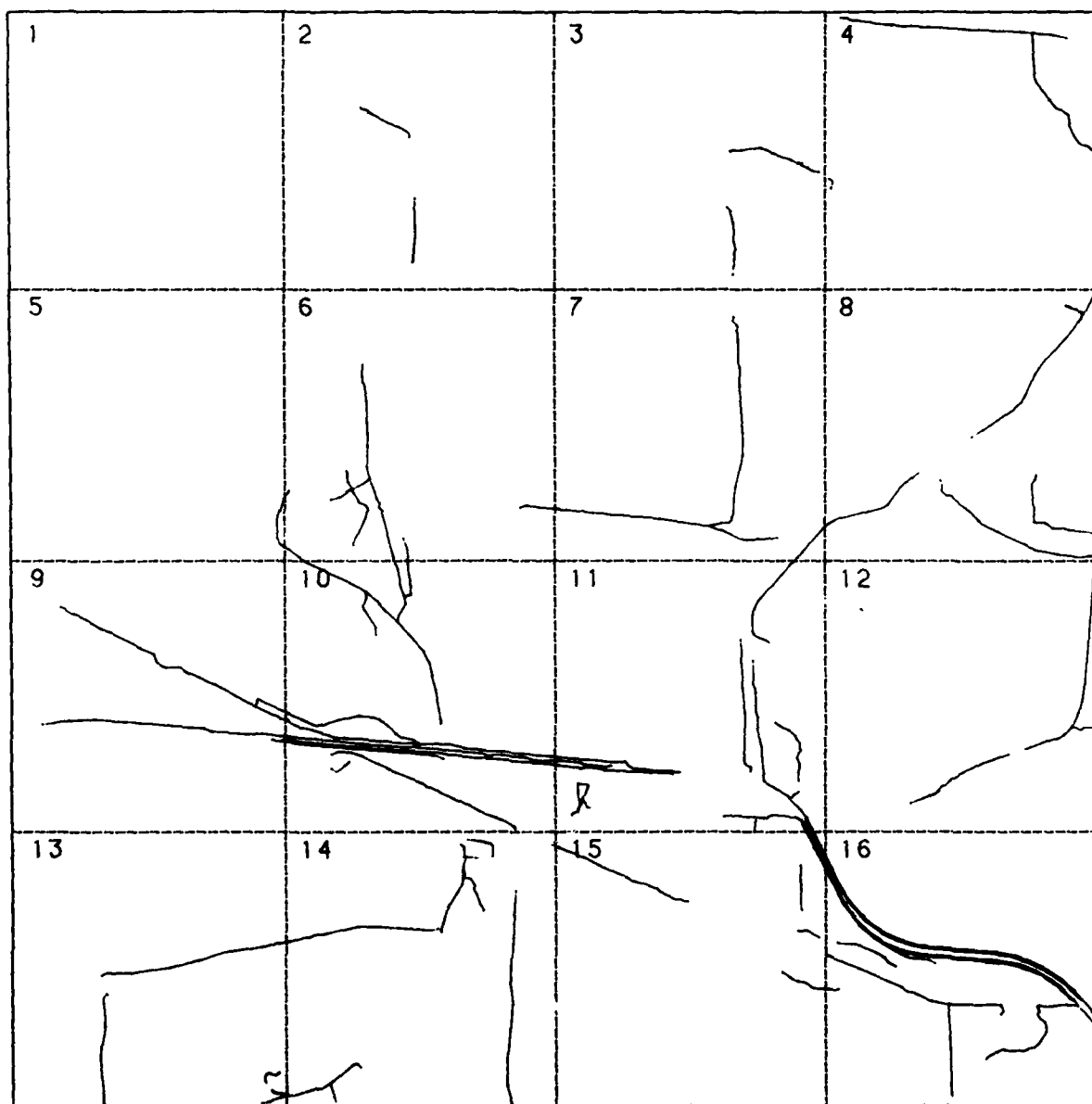


Fig. 5.30: Result of initial extraction from image No. 56
as shown in fig. 5.16

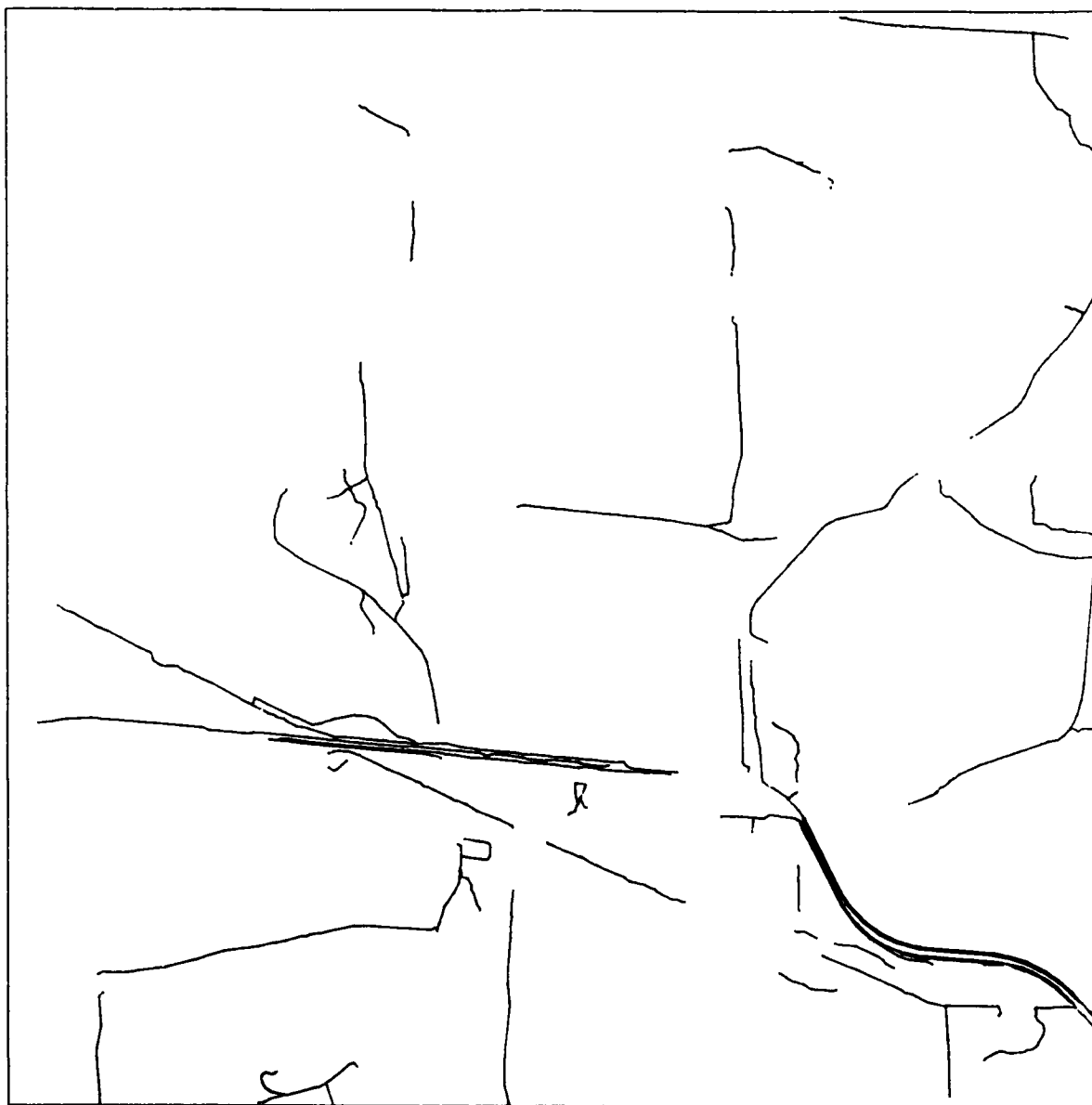


Fig. 5.31: Result of maximum length polygon compilation

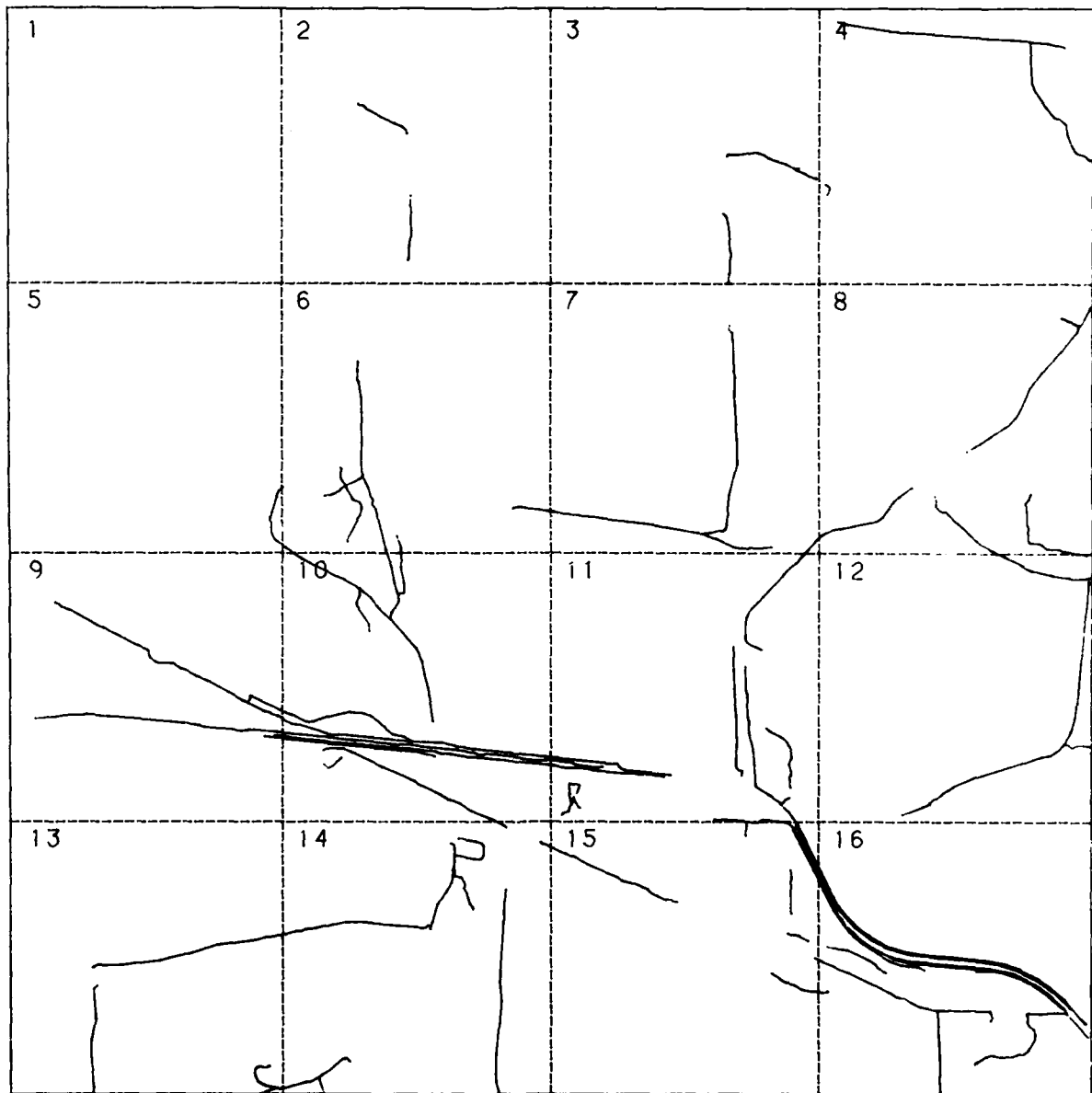


Fig. 5.32: Result of mapping the maximum length polygons of
fig. 5.31 to the coordinate system of image No. 54

- fig. 5.33 shows the result of the verification in image No. 54
- fig. 5.34 shows the result of the first continuation extraction from image No. 54 which generates a significant amount of new lines
- the color print of fig. 5.35 shows a composite of the verification result (fig. 5.33) in yellow and the continuation extraction result (fig. 5.34) in blue. The completion effect is obvious
- the color print of fig. 5.36 shows, for image No. 56, a composite of the accepted part of the initial extraction result (in yellow) and the accepted part of the first continuation extraction result (in violet) which came from image No. 54 and was used, after applying the mapping function, to guide the second verification process
- fig. 5.37 shows the result of the second continuation extraction which works on the data of image No. 56. This result is submitted to verification in image No. 54; the presentation of this operation's result is omitted
- fig. 5.38 and 5.39 show the final results for image No. 56 and No. 54 respectively. In both figures the yellow part of the line network resulted from extraction processing in image No. 56 and verification in image No. 54, the blue part of the line network resulted from the extraction process in image No. 54 with verification in image No. 56
- fig. 5.40 shows the result of a third continuation extraction from image No. 54, which does not contribute significantly to the final result, and, hence, was used as a reason to stop the whole procedure of extraction and verification.

Comparing the final results of both experiments, i.e. fig. 5.29 to fig. 5.38 and fig. 5.28 to fig. 5.39, one can state that, in both cases, the basic line network is identical although a different sequence of processing steps has been applied. This clearly proves the flexibility and - to a considerable degree - the data independence of our procedure. The final results are much more complete than single, initial extraction results and in addition they reflect local deviations which are inherent to corresponding images of objects in stereo pairs.



Fig. 5.33: Result of the first verification performed
in image No. 54

AD-A133 892

COINCIDENT EXTRACTION OF LINE OBJECTS FROM STEREO IMAGE

2/2

PAIRS(U) FORSCHUNGSINSTITUT FUER

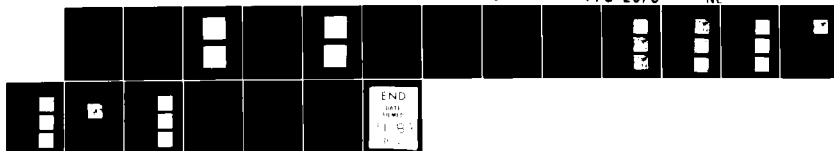
INFORMATIONSVARBEITUNG KARLSRUHE (G..

UNCLASSIFIED

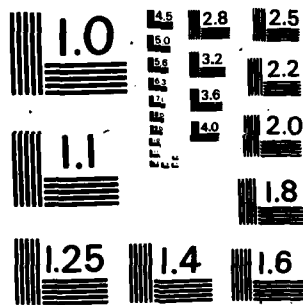
H KAZMIERCZAK ET AL. SEP 83 FIM-115

F/G 20/6

NL



END
DATE
FILMED
11 8 83
GPO



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A



Fig. 5.34: Result of the first continuation extraction
performed in image No. 54

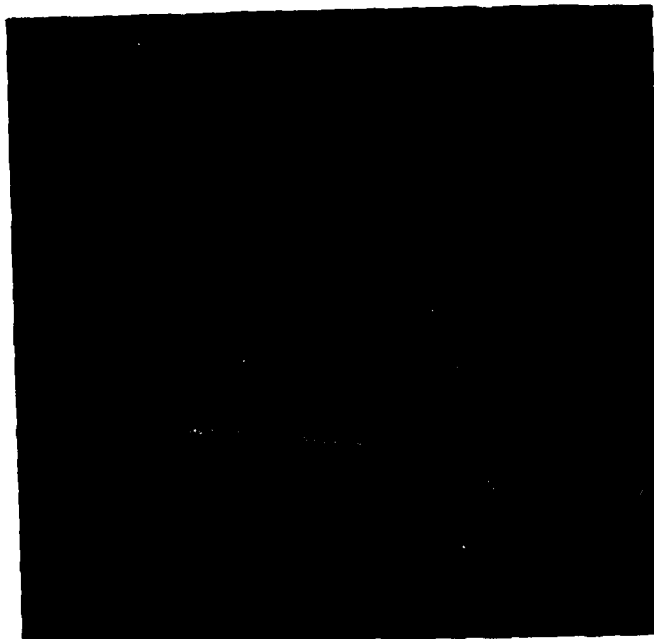


Fig. 5.35: Composite of the result of the first verification (yellow) and the result of the first continuation extraction (blue)



Fig. 5.36: Composite of the result of the first verification (yellow) and the result of the second verification (violet) related to image No. 56

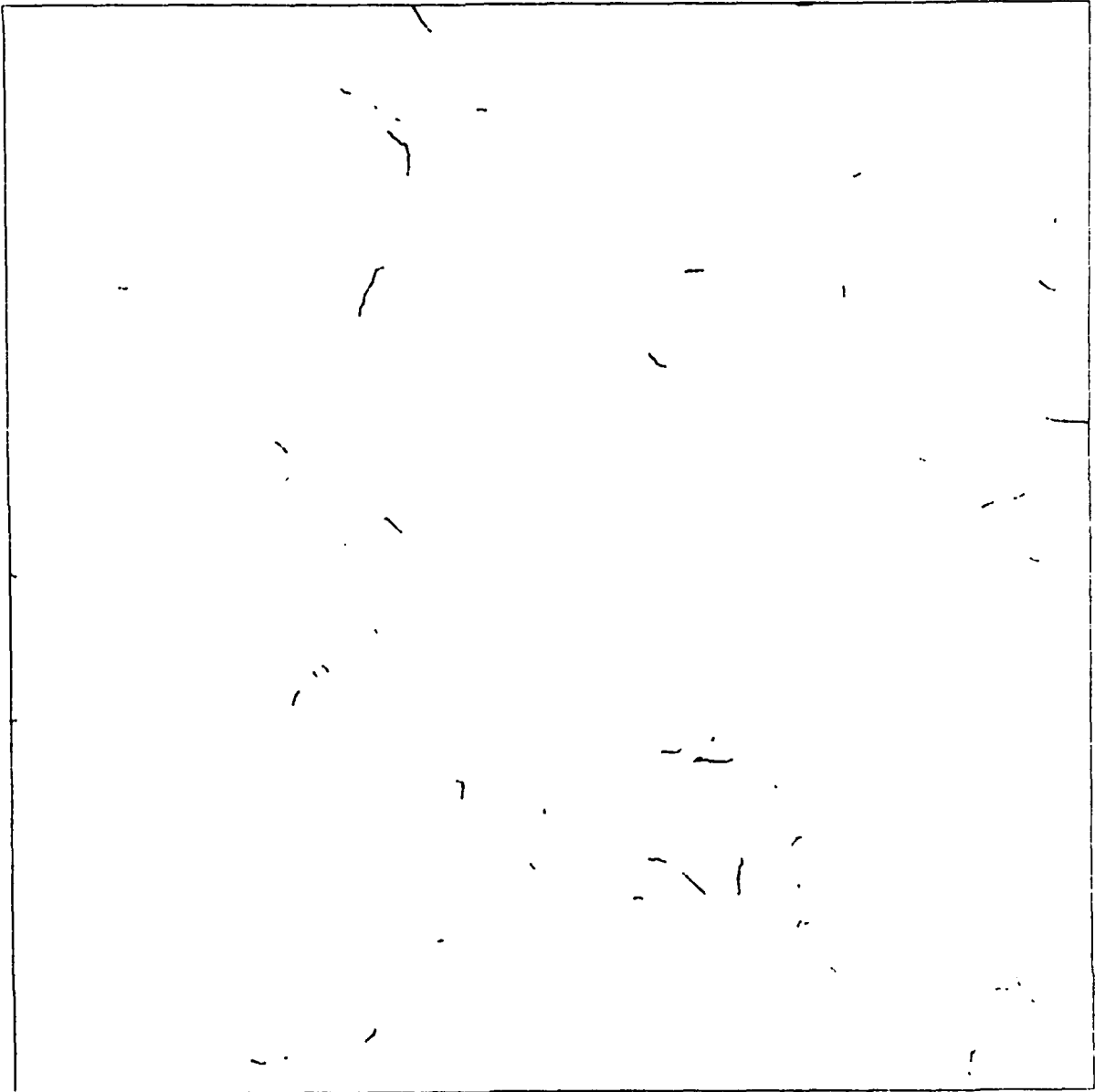


Fig. 5.37: Result of the second continuation extraction
performed in image No. 56

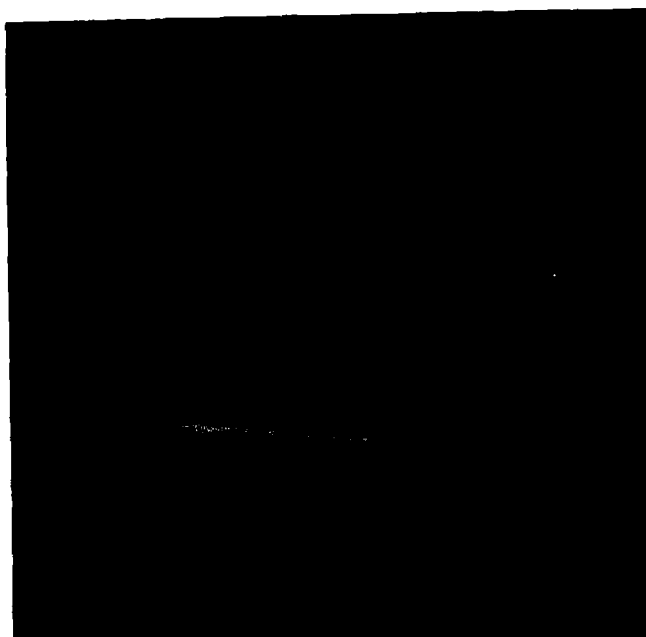


Fig. 5.38: Final result for image No. 56 consisting of extraction results from image No. 56 (yellow) and prediction results from image No. 54 (blue)

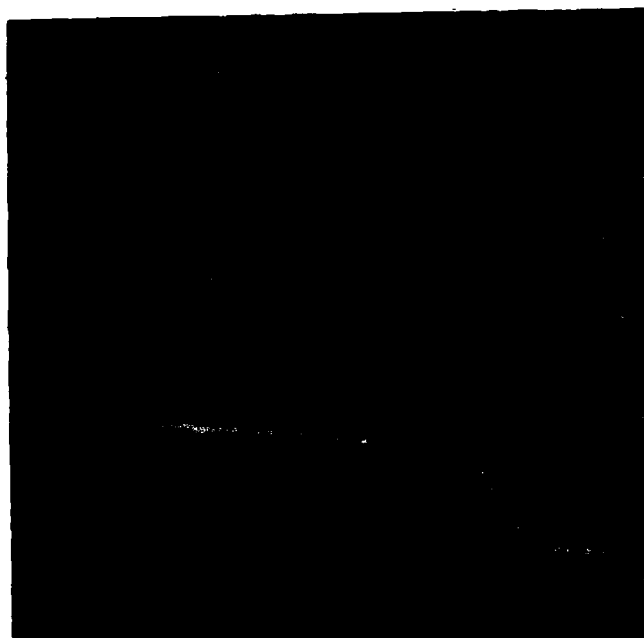


Fig. 5.39: Final result for image No. 54 consisting of prediction results from image No. 56 (yellow) and extraction results from image No. 54 (blue)

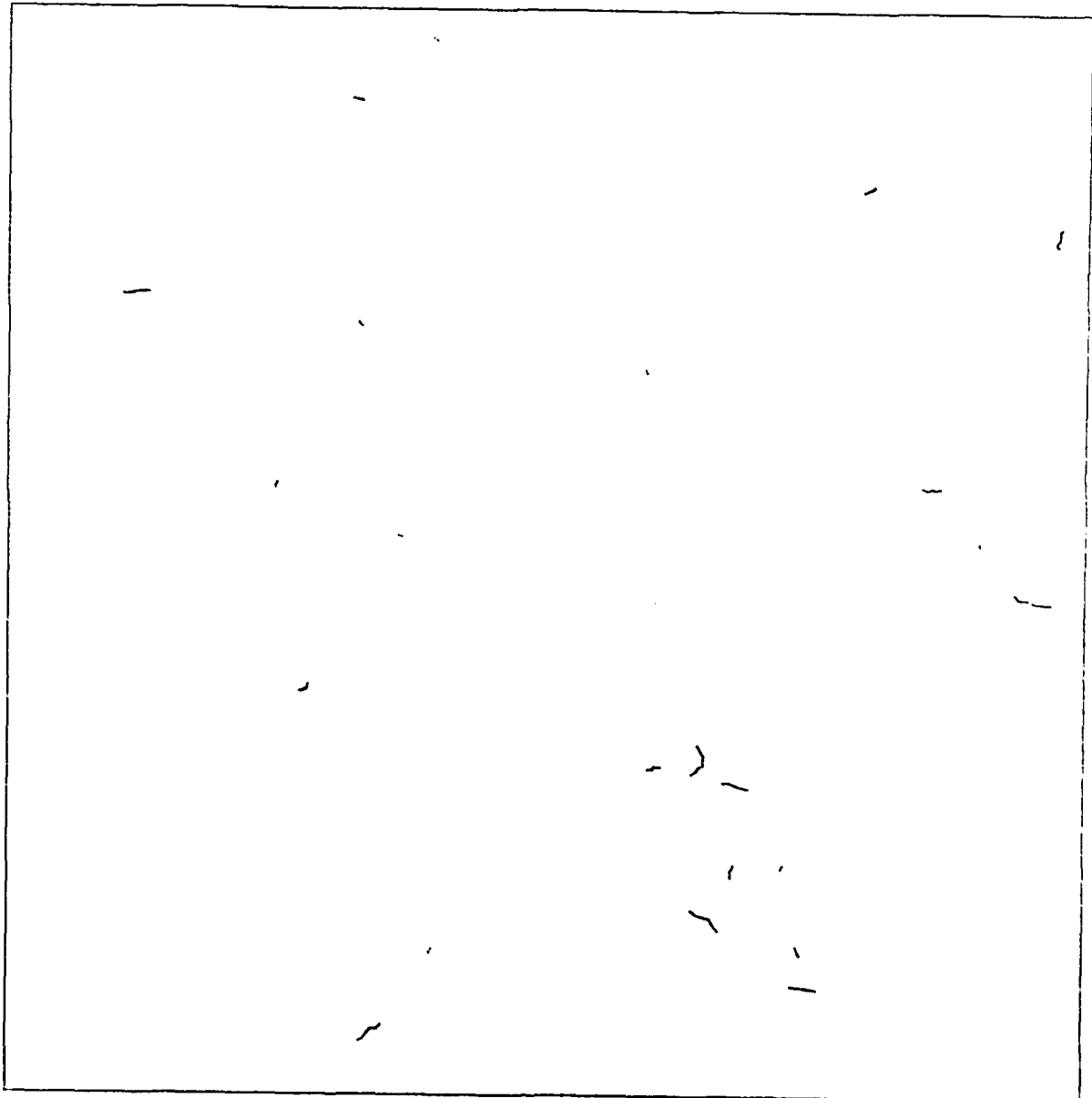


Fig. 5.40: Result of the third continuation extraction performed in image No. 54, which is not added to the final results

Table 5.3 indicates the processing times of the different steps of the second experiment. They compare quite well to the respective times of table 5.2. In the following section, details of the results will be shown and explained at a larger scale.

Process	CPU time	Elapsed time
Initial extraction	70	145
First MLP compilation	40	165
First mapping	0.1	0.3
First verification	120	405
First continuation extraction	55	105
Second MLP compilation	20	80
Second mapping	0.1	0.2
Second verification	90	330
Second continuation extraction	20	55
Third MLP compilation	15	55
Third mapping	0.1	0.2
Third verification	50	195
Third continuation extraction	15	45
Final result processing	115	420

Table 5.3: Processing times for coincident extraction starting from image No. 56 (approximate time in minutes)

5.4 Explanation of details in segments of the stereo image pairs of 1000 x 1000 pixels

Details of the procedure of coincident line extraction will be explained in different segments of the stereo image pair No. 54 and No. 56. We reduced the size of image segments by a factor of four to generate 512 x 512 pixel displays, overlaid the respective graphic result and photographed the COMTAL screen. The pictures, some of which are color pictures, are shown in the following figures.

Figure 5.41 shows the image data of segment No. 8 of image No. 56 as shown in fig. 5.16. Figure 5.42 shows the result of an initial extraction from that image segment: there are four starting points indicated as white dots; that starting point in the bottom right part of the figure clearly does not mark a section of a road, but a track in a field; however, as the form of this track complies completely to the line objects we specified by the extraction parameter set, the respective starting point is accepted to be valid. At each of the four starting points LEM begins the extraction operation; its result is shown in red color. When LEM stops, REM takes over; the result of REM is shown in yellow color. Note that the vertical road of the left part and the two horizontal roads in the middle part of the figure are not found during the initial extraction run.

Fig. 5.43 shows the composite of the corresponding segment of image No. 54 with the mapped initial extraction result. The local accuracy of the mapping function is excellent in this example. Fig. 5.44 shows the result of the verification process (see also fig. 5.33). Some smaller parts of the predicted network of lines have been rejected by the verification. Fig. 5.45 shows the composite of the verification result and the first continuation extraction result from this segment of image No. 54. Note that the majority of missing roads has now

been extracted (but must still be verified by evaluation of the corresponding segment parts of image No. 56). Fig. 5.46 shows the result of this second verification process, together with the result of the first verification process after mapping back to the coordinate system of this image. The second continuation extraction adds some more small parts of the lines to complete the network (see fig. 5.47), at least preliminary, because the continuations have to be verified, still.

The last color photographs of this sequence (figures 5.48 and 5.49) show the final results generated for the two corresponding segments No. 8 of the stereo pair of images No. 54 and 56. These are overlays of the image data with the final result matrix contents which consists of symbols I (for internal, shown in yellow color in each figure) and symbols E (for external, shown in red color in each figure). Compare figures 5.38 and 5.39 respectively for the context. That part of the road covered by trees in the middle of the segment could not be extracted, two small parts of the roads also hidden by trees (a vertical part at the left side and a horizontal part in the middle) have not been found to be acceptable. Small access roads to farm buildings and into the fields have been found.

To complete the description of details we present the result of the complementary extraction experiment which started the initial extraction from image No. 54. Fig. 5.50 shows the image data of segment No. 8 of image No. 54. The result of the initial extraction is shown in fig. 5.51 in color: white dots indicate starting points; note that there are more and other starting points than in the corresponding segment of image No. 56 (fig. 5.42); red colored lines stand for the extraction result of LEM, yellow colored lines stand for the extraction result of REM. Note that the segment No. 8 of image No. 54 obviously is of better image quality, because the initial extraction result is much more complete than the respective result from image No. 56.

Fig. 5.41: Segment No. 8 of
fig. 5.16

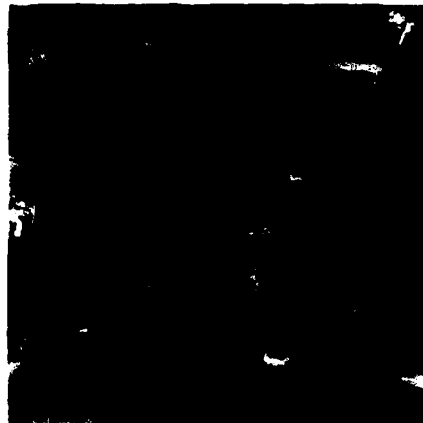


Fig. 5.43: Mapping of the
initial extraction result
to segment No. 8 of fig. 5.15



Fig. 5.44: Result of the
first verification in seg-
ment No. 8 of No. 54



Fig. 5.45: Update by the
first continuation extrac-
tion result



Fig. 5.46: Verification of
the first continuation ex-
traction result



Fig. 5.47: Update by the
second continuation ex-
traction result



Fig. 5.42: Result of initial extraction from segment No. 8 as shown in fig. 5.41 containing starting points (white dots), LEM results (red lines) and REM results (yellow lines)



Fig. 5.48: Final result for segment No. 8 of image No. 54 as shown in fig. 5.15: extraction results from No. 54 in yellow, prediction results from No. 56 in red



Fig. 5.49: Final result for segment No. 8 of image No. 56 as shown in fig. 5.16: extraction results from No. 56 in yellow, prediction results from No. 54 in red



Fig. 5.50: Segment No. 8 of
image No. 54 as
shown in fig. 5.15



The results of the different steps of the procedure are omitted here. The next two figures 5.52 and 5.53 show, in color, the final results of the complete procedure. Here again, yellow marks the results extracted from the respective segment and verified in the corresponding one (symbol I of the result matrix) and the red color stands for results extracted from the corresponding segment and verified in the respective one. It is evident that the basic pattern of the results is identical to the results shown in figures 5.49 and 5.48 respectively.

To conclude the presentation of results, we show details of segment No. 7 of the experiment which started the extraction from image No. 56. Fig. 5.54 shows the image data of that segment. Fig. 5.55 shows the result of the initial extraction, coded by the colors white (starting points), red (LEM result) and yellow (REM result). Note the two starting points in the upper part of the segment which were not accepted by LEM during this initial extraction step. The following two figures 5.56 and 5.57 show the final results for this segment of image No. 56 and image No. 54 respectively. The color code of the final result has already been explained: yellow represents the symbol I of the result matrix, red stands for the symbol E of the

Fig. 5.51: Result of initial extraction from segment No. 8 as shown in fig. 5.50 containing starting points (white dots), LEM results (red) and REM results (yellow)



Fig. 5.52: Final result for segment No. 8 of image No. 54, when starting the extraction procedure in this image: extraction results from No. 54 in yellow, prediction results from No. 56 in red



Fig. 5.53: Complementary final result for segment No. 8 of image No. 56, when starting the extraction procedure in image No. 54: extraction results from No. 56 in yellow, prediction results from No. 54 in red



result matrix. Regarding this example, it can be seen that only very few continuing parts of the line network were added to the initial extraction result by the various extraction and verification steps of the whole procedure.



Fig. 5.54: Segment No. 7 of image No. 56 as shown in fig. 5.16

Fig. 5.55: Result of the initial extraction from segment No. 7 as shown in fig. 5.54 (see fig. 5.42 for color explanation)

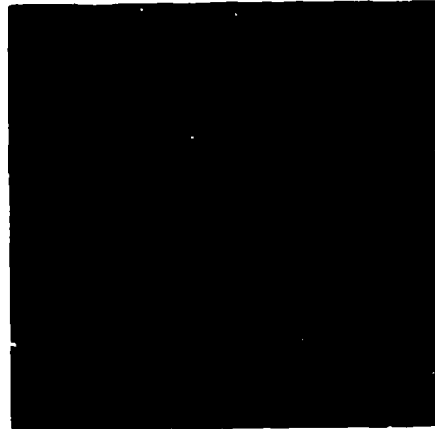


Fig. 5.56: Final result for segment No. 7 of image No. 56 (see fig. 5.48 for color explanation)

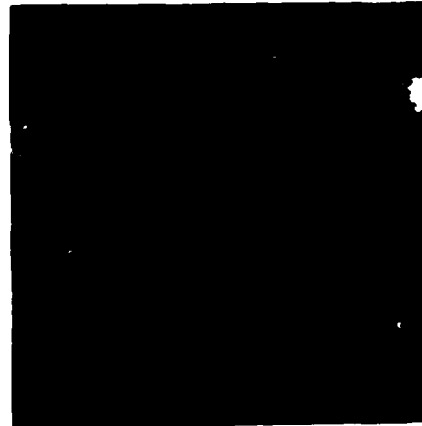


Fig. 5.57: Complementary final result for segment No. 7 of image No. 54 as shown in fig. 5.15 (see fig. 5.49 for color explanation)



6. Discussion of the results and conclusions

It has been shown that the development and implementation of the procedure for coincident line extraction from aerial image stereo pairs has solved two problems:

- the extraction result is more complete and connected to a higher degree, when compared to the extraction result from a single image
- corresponding results are generated for line objects contained in both images of a stereo pair. The generation of these results is independent of the external orientation of the imaging devices and, thus, the results represent information intrinsic to the respective image.

The implementation of the total extraction procedure was not optimized with respect to processing time and storage requirements. In fact, the various measurements indicate considerable processing time requirements. However, the measurements also prove that, within our implementation, an average of 50% of the total CPU time is used for processing of image and result data transfer to and from disk. The image data evaluation itself does not require tremendous amounts of CPU time. On the other hand, the unbelievable big difference between CPU time and total elapsed time simply is caused by waiting for data transfer. To our opinion, there is still a considerable optimization potential with respect to data organization and processing time.

The methods work successfully on image data of differing image quality; the methods are especially suited to evaluate images of rural terrain and urban areas without dense population. A similar evaluation of images of densely populated areas is an unsolved problem, still, which needs a much higher degree of sophistication.

Acknowledgement

The authors gratefully appreciate the valuable help of Dipl.-Ing. Peter Fritsche, Dr.-Ing. Wolf-Dieter Groch, Dipl.-Inf. Helga Mayer and Dipl.-Inf. Ruben Neu who contributed significantly to develop and implement the system and to produce the results and parts of the report.

References

- /1/ U. Bausch et al. Automatic Extraction of Linear Features from Aerial Photographs Final Technical Report for ERO London, April 1981
- /2/ M. Bohner et al. Automatic Extraction of Linear Features from Aerial Photographs Final Supplemental Technical Report for ERO, London, April 1981
- /3/ M. Sties Third Progress Report on Coincident Extraction of Line Objects From Stereo Image Pairs, Oct. 1982
- /4/ H. Niemann Pattern Analysis Springer, Berlin, 1981
- /5/ W. Göpfert Methodology for Thematic Image Processing Using Thematic and Topographic Data Bases and Base-Integrated Multi-Sensor Imagery, Actes du Symposium International de la Commission VII de la Société Internationale de Photogrammétrie et Télédétection, Toulouse 1982, Volume 1, pp. 13-19

END

DATE
FILMED

11 83

DT